



QuickStart Manual

QS-DSOFT32-M



WARNING



Thank you for purchasing automation equipment from Automationdirect.com. We want your new automation equipment to operate safely. Anyone who installs or uses this equipment should read this publication (and any other relevant publications) before installing or operating the equipment.

To minimize the risk of potential safety problems, you should follow all applicable local and national codes that regulate the installation and operation of your equipment. These codes vary from area to area and usually change with time. It is your responsibility to determine which codes should be followed, and to verify that the equipment, installation, and operation is in compliance with the latest revision of these codes.

At a minimum, you should follow all applicable sections of the National Fire Code, National Electrical Code, and the codes of the National Electrical Manufacturer's Association (NEMA). There may be local regulatory or government offices that can also help determine which codes and standards are necessary for safe installation and operation.

Equipment damage or serious injury to personnel can result from the failure to follow all applicable codes and standards. We do not guarantee the products described in this publication are suitable for your particular application, nor do we assume any responsibility for your product design, installation, or operation.

If you have any questions concerning the installation or operation of this equipment, or if you need additional information, please call us at 1-770-844-4200.

This publication is based on information that was available at the time it was printed. At Automationdirect.com we constantly strive to improve our products and services, so we reserve the right to make changes to the products and/or publications at any time without notice and without any obligation. This publication may also discuss features that may not be available in certain revisions of the product.

Trademarks

This publication may contain references to products produced and/or offered by other companies. The product and company names may be trademarked and are the sole property of their respective owners. Automationdirect.com disclaims any proprietary interest in the marks and names of others.

Stage is a trademark of Koyo Electronics Industries Co., LTD. Texas Instruments is a registered trademark of Texas Instruments, Inc. TI, TIWAY, Series 305, Series 405, TI305, and TI405 are trademarks of Texas Instruments, Inc. Siemens and SIMATIC are registered trademarks of Siemens, AG. GE is a registered trademark of General Electric Corporation. Series One is a registered trademark of GE Fanuc Automation North America, Inc. MODBUS is a registered trademark of Gould, Inc. IBM is a registered trademark of International Business Machines. MS-DOS and Microsoft are registered trademarks of Microsoft Corporation. Windows is a trademark of Microsoft Corporation. OPTOMUX and PAMUX are trademarks of OPTO 22.

**Copyright 1999, Automationdirect.com Incorporated
All Rights Reserved**

No part of this manual shall be copied, reproduced, or transmitted in any way without the prior, written consent of Automationdirect.com Incorporated. Automationdirect.com retains the exclusive rights to all information included in this document.

Manual Revisions

If you contact us in reference to this manual, remember to include the revision number.

Title: DirectSOFT Quick Start User Manual

Manual Number: QS-DSOFT32-M

Issue	Date	Effective Pages	Description of Changes
Original	9/96	Cover/Copyright Contents Manual Revisions 1 — 57	Original Issue
2nd Edition	2/97	Contents Manual Revisions 1 — 56	Down size format
3rd Edition	6/98	Contents Manual Revisions 1 — 56	Add D3-350 Release 2.3 (3 diskettes)
4th Edition	8/99	Contents Manual Revisions 1 — 56	Release 3.0, 32-bit application (CD)

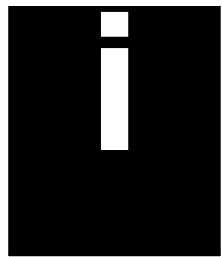


Table of Contents

Introduction	1
The Purpose of this Supplementary Manual	1
Who Can and Should Use DirectSOFT?	1
Preparing for Installation	2
Getting to Know Windows	2
Check Your PC Hardware Requirements	2
Power Supply	2
DirectSOFT32 package contents	2
PLC Compatibility	3
Supported Devices	3
Installation of DirectSOFT32 Software	4
Step 1: Load the CD	4
Step 2: Review your options	4
Step 3: Enter the Security Code	4
Step 4: Unpacking the software	4
Step 5: End Other Windows Tasks	5
Step 6: <i>Direct</i> SOFT32 License Agreement	5
Step 7: Enter Your Name and Company	5
Step 8: Select Installation Directory	6
Step 9: Select Installation Type	6
Step 10: Custom Installation	6
Step 11: Program Installation	7
Step 12: Program Verification	7
Building an Example Program	8
What You Will Learn	8
Example	8
Ladder Logic Example	8
Step 1: Enter the Programming Mode	9
Step 2: Start a New Project	9
Step 3: Switch to the Edit Mode	10
Step 4: Using the Ladder Palette to Enter the First Element	11
Step 5: Using the SP0 Relay in Our Program	12
Step 6: Using the Box Browser	13
Step 7: Scrolling the Box Class Window	13
Step 8: Entering a Preset in the Accumulator	14
Step 9: Connecting Elements in Parallel	15
Step 10: Inserting an Output Box	15
Step 11: Entering the Address to Hold the Preset	16
Step 12: Enabling the Documentation Options	17
Step 13: Entering a Comment	17
Step 14: Assigning a Nickname to the Preset Output	19
Step 15: Adding an Internal Relay to Start the Timer	21
Step 16: Assigning a Nickname to the Start Contact	22
Step 17: Adding a Normally Closed Contact	23
Step 18: Adding a Nickname to the Normally Closed Contact	23
Step 19: Inserting the Timer Instruction	24
Step 20: Adding the Counter	26
Step 21: Entering the Counter Data	27
Step 22: Assigning a Nickname to the Counter	27
Step 23: Making the Counter Self-Resetting	28

Step 24: Adding a Contact for Resetting on the First Scan	30
Step 25: Documenting the Function of the Counter	32
Step 26: Adding a Comparative Boolean Instruction	33
Step 27: Adding a Conditional Output	33
Step 28: Copying to the Clipboard	35
Step 29: Pasting to your Program	36
Step 30: Ending the Program	38
Connecting Online to a PLC & Downloading the Program	39
Create a Standard Serial Link	39
Link Status	42
Writing your Program to the PLC	43
Monitoring the Program with a Data View	44
Creating a Data View	44
Adding Dynamic ON/OFF Edit Buttons	45
Entering the Run Mode	46
Observing the Status of Elements	47
Using the Edit Buttons to Change Status	47
Writing Edits to the PLC	48
Special Data View Instructions for DL305 Family	49
DirectNET Restrictions	49
Changing the Value to Binary to Observe All 8 Bits	50
Changing to the Run Mode	51
Enabling the All Status ON	51
Entering Data for the Status Register	51
Writing the Edits to the PLC	52
Troubleshooting Guide	53
Communication Problems	53
Comm Error Connecting to PLC	54
Internal Modem Card Conflicts	54
Mouse Driver Conflicts	54
Swapping Ports to Solve a Mouse Conflict	55
Other Driver Conflicts	55
Driver Conflicts with Laptop Computers	55
Power Management Conflicts	56
Specialized Video Device Driver Problems	56
Conflicts with other PLC Vendor Software Drivers	56
Non-Shunted Power Supplies	56
Screen Saver Conflicts	56
Printing Problems	57
Programming Cables	57

Introduction

The Purpose of this Supplementary Manual

With this Quick Start manual, you should learn enough of the basics to get started without having to read the Users manual that covers all of the details of **Direct**SOFT32. If you have received this with your **Direct**SOFT demo disk, it will provide you with a sampling of how easy **Direct**SOFT32 is to use. For those who have purchased the full version of **Direct**SOFT32, exercise caution: **This is not intended to replace reading the Users Manual. This is intended only as a supplement. This is only a quick start!**

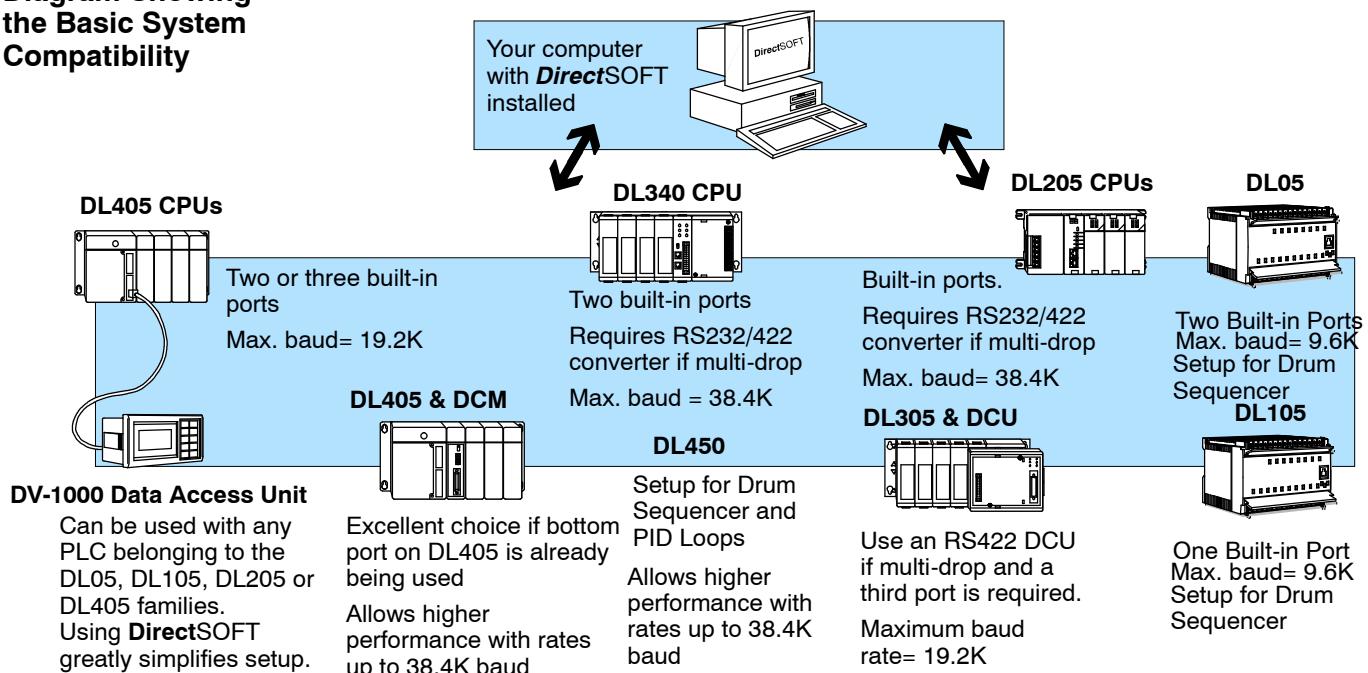
Who Can and Should Use **Direct**SOFT?

If you have a PLC belonging to the **Direct**LOGIC CPU family, you can use **Direct**SOFT to create your ladder logic programs. The families of PLCs (DL05, DL105, DL205, DL305 and DL405) that currently exist under this description are shown below. We have included some other useful and related information. **Direct**SOFT32 has added the following functionality;

- set up a DV1000 Data Access Unit
- tune PID loops for the D2-250, D3-350, and D4-450
- set up the parameters for Drum Sequencers in the DL05, DL105, D2-250, D3-350, and D4-450

DirectSOFT will also work with many **Direct**LOGIC compatible products (not shown in the diagram). If you fall into this category, however, the chart on the next page shows you a complete list of which products work with the software.

Diagram Showing the Basic System Compatibility



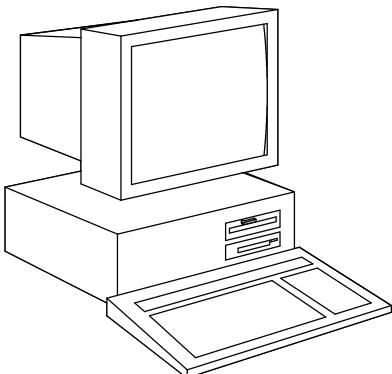
Preparing for Installation

Getting to Know Windows

DirectSOFT32 software runs under 32-bit Windows operating systems (95/98/NT). If you are more accustomed to using Windows 3.1, please take a moment to study your PC's reference manual on the operation of Windows 95/98/NT.

Check Your PC Hardware Requirements

Please check the following requirements when choosing your PC configuration.



System Requirements

- Windows 95/98/NT-compatible Pentium 133 (or higher)
- 32Mb RAM
- 11Mb available hard drive disk space
- CD-ROM drive
- Color monitor, (640x480 or greater)



NOTE on Laptops: **Direct**SOFT32 is perfectly suitable for use with laptop computers as long as they meet the requirements shown above. However, there can be a few problem areas. One inconvenient problem is when your laptop only has one COM port and does not have a built-in mouse. In this case, you will have to use the COM port for the PLC communications instead of the mouse connection.



Tip on Monitors: Any size monitor will work, but larger monitors enhance the display capabilities of **Direct**SOFT32. Also, we strongly recommend a color monitor. **Direct**SOFT32 makes use of color to note certain conditions, such as program editing changes, error conditions, etc. It will work with monochrome monitors, but the results are enhanced with color monitors.

Power Supply

It is highly recommended that the computer **Direct**SOFT32 operates on has some form of power surge protection. A quality surge protector will protect your computer from most surges and spikes however, an uninterruptible power supply (UPS) will provide the ultimate protection. A UPS provides complete isolation between the AC power source and the computer and has battery backup for blackout and brownout conditions.

DirectSOFT32 package contents

Now is the time to review the contents of your **Direct**SOFT32 software package. You should have the following items:

- CD ROM
- Quick Start Manual
- Learning Guide Manual
- License Agreement
- Registration Card
- Thank You Letter
- Feedback Form

PLC Compatibility

Family	CPU	<i>Direct-SOFT Programming</i> PC-PGMSW	<i>Direct-SOFT Programming</i> Single Family	<i>Direct-SOFT Site Licenses</i>	<i>Direct-SOFT OEM License</i>	<i>Direct-SOFT DDE Server</i>
DL05	(requires Rel. 2.4 or later)	✓	PC-PGM105	✓		✓
DL105	F1-130** (requires Rel. 2.0 or later)	✓	PC-PGM105	✓		✓
DL205	D2-230	✓	PC-PGM205	✓	PC-D2OEM	✓
	D2-240	✓	PC-PGM205	✓	PC-D2OEM	✓
	D2-250 (requires Rel. 2.1 or later)	✓	PC-PGM205	✓	PC-D2OEM	✓
DL305	D3-330*, D3-330P*	✓		✓	PC-D3OEM	✓
	D3-340	✓		✓	PC-D3OEM	✓
	D3-350 (requires Rel. 2.3 or later)	✓	PC-PGM350	✓	PC-D3OEM	✓
DL405	D4-430	✓		✓	PC-D4OEM	✓
	D4-440**	✓		✓	PC-D4OEM	✓
	D4-450** (requires Rel. 2.0 or later)	✓		✓	PC-D4OEM	✓
GE® Series 1	IC610CPU105*	✓		✓	PC-D3OEM	✓
	IC610CPU106*	✓		✓	PC-D3OEM	✓
TI305™ / SIMATIC® TI305™	325-07*, PPX:325-07*	✓		✓	PC-D3OEM	✓
	330-37*, PPX:330-37*	✓		✓	PC-D3OEM	✓
	325S-07* (or 325 with Stage Kit)	✓		✓	PC-D3OEM	✓
	330S-37*, PPX:330S-37*	✓		✓	PC-D3OEM	✓
	335-37, PPX:335-37	✓		✓	PC-D3OEM	✓
TI405™ / SIMATIC® TI405™	425-CPU, PPX:425-CPU **	✓		✓	PC-D4OEM	✓
	PPX:430-CPU	✓		✓	PC-D4OEM	✓
	435-CPU, PPX:435-CPU **	✓		✓	PC-D4OEM	✓

* — requires Data Communications Unit (D3-232-DCU) ** — also DC versions

NOTE: In general, the compatible products listed offer similar features and are even identical in some cases. However, DirectSOFT32 has not been completely tested with the compatible products. There may be some aspects of system operation that may not be supported, or, that may not work the same as previous software packages.

Supported Devices One of the benefits with the *DirectLOGIC* family is the wide variety of programming connections. For example, you can use *DirectSOFT32* to communicate directly with a PLC, or, you can use a communications device, such as the DL405 Data Communications Module. Below is a list of supported devices:

Data Communication Devices:

- DL405 Data Communications Module (D4-DCM)
- DL405 Ethernet Communications Module (H4-ECOM)
- DL305 Data Communications Unit (D3-232-DCU, D3-422-DCU)
- DL205 Data Communications Module (D2-DCM)
- DL205 Ethernet Communications Module (D2-ECOM)

I/O Modules:

- DL405 Slice I/O Modules
- D2-RSSS (Slice protocol)

Installation of *DirectSOFT32* Software

Step 1:
Load the CD.

Step 2:
Review your options

The CD will begin its auto-install feature. This CD also contains demo versions of our other software products. You will see a screen that provides you with different options. You can browse the CD or install a program. When you are ready to install the program, click on that selection.



Step 3:
Enter the Security Code

This software package is protected by a security code. The code is located on the outside of the CD jewel case. Complete the information in the registration window and click the **Finish** button.

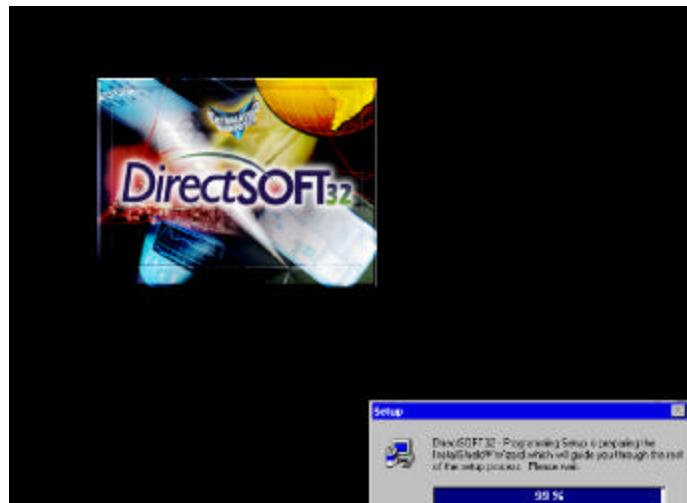
Note: You must enter the password exactly as it appears (dashes, spaces, capital letters, etc).

If you have entered the number incorrectly, the **Finish** button will not be accessible..



Step 4:
Unpacking the software

The installation process begins by unpacking the information on the CD that corresponds to the security code entered. Pop-up windows will show you the status of the unpacking.



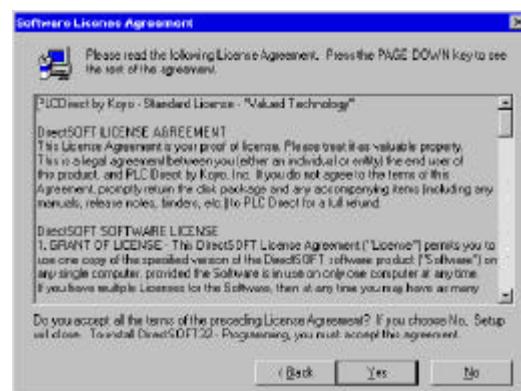
Step 5: End Other Windows Tasks

The installation issues a reminder to exit all other Windows applications. If you are unsure of the programs that may be running, press Ctrl-Alt-Delete and select the Task Manager. If everything is closed, click *Next* to continue.



Step 6: DirectSOFT32 License Agreement

The next screen displays the software license agreement. If you agree to the terms and conditions, click *Yes* to continue.



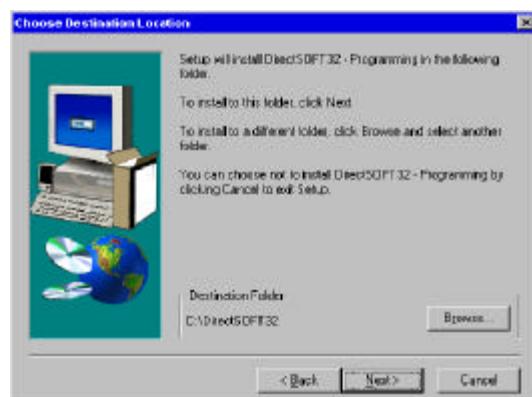
Step 7: Enter Your Name and Company

The next information required to enter is your name and the name of your company. This will register this copy to you.



**Step 8:
Select Installation
Directory**

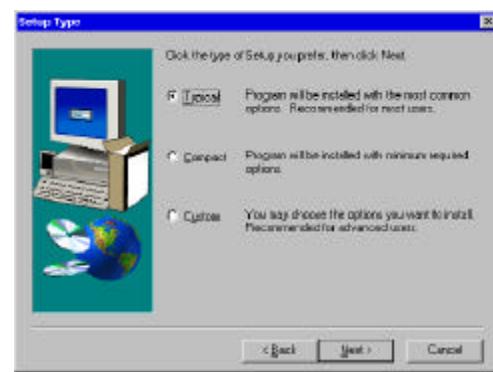
The Select Program Folder dialog lets you choose the folder (directory) in which the **Direct**SOFT32 files will be loaded.



**Step 9:
Select Installation
Type**

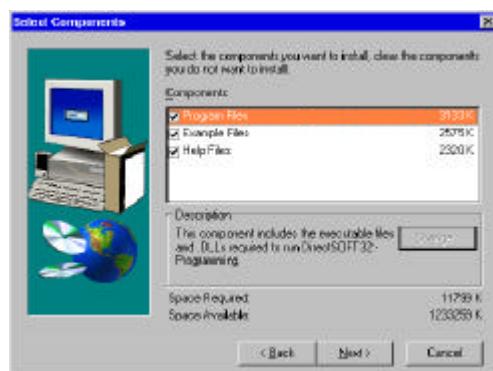
You now have an option of what type of install that you want. The *Typical* install loads everything **Direct**SOFT32 has to offer (Program Files, Example Files, Help Files, Files). The *Custom* option lets you choose which features to install. The *Compact* option installs the Program Files only.

Generally you will choose the *Typical* installation.



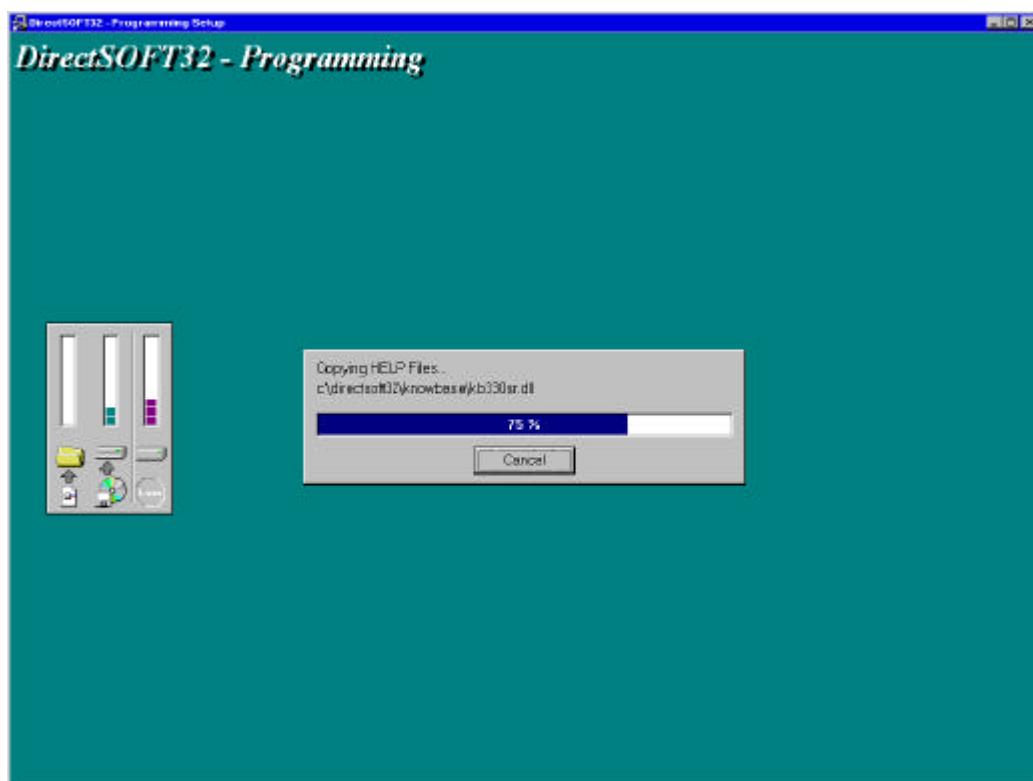
**Step 10:
Custom
Installation**

If you choose the *Custom* installation you will be prompted to select what features to install.



**Step 11:
Program
Installation**

With all of the options selected and the parameters completed, the program can begin its installation. The screen provides information on the status of the install.

**Step 12:
Program
Verification**

After the program has successfully installed, a screen may appear with verification and ask you to restart Windows. If this occurs, you must restart Windows before you can use the program.,.



Building an Example Program

What You Will Learn

The pages that follow will explain how to do the following:

- create a project
- use the tool palette to enter instructions and build a ladder program
- use “hot keys” to work faster
- enter nicknames and add comments
- setup an internal timer
- setup a self-resetting internal counter
- cut and paste rungs of logic
- save your program to disk
- communicate with your PLC
- load your program into PLC memory
- monitor your program with the Data View



NOTE: The following program is given only to illustrate how some of the key features of the software operate. This is not intended to teach you how to write ladder logic.

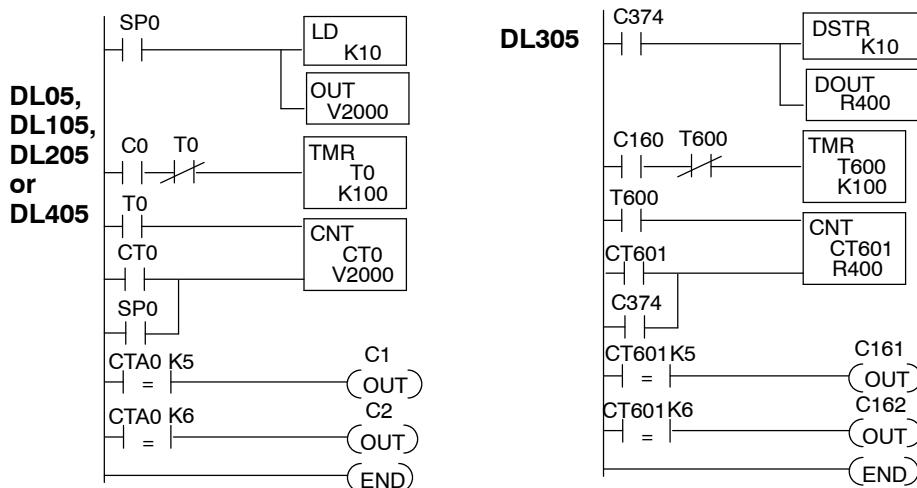
Example

This example has four basic tasks:

1. Load a value into memory of your PLC that can be used as a preset for a timer.
2. Setup a self-resetting timer.
3. Use a counter to count the number of times the timer reaches the preset value and resets.
4. Use Comparative Boolean relays to turn ON an internal coil when the counter current value equals 5, and turn ON a second internal coil when the counter current value equals 6.

Ladder Logic Example

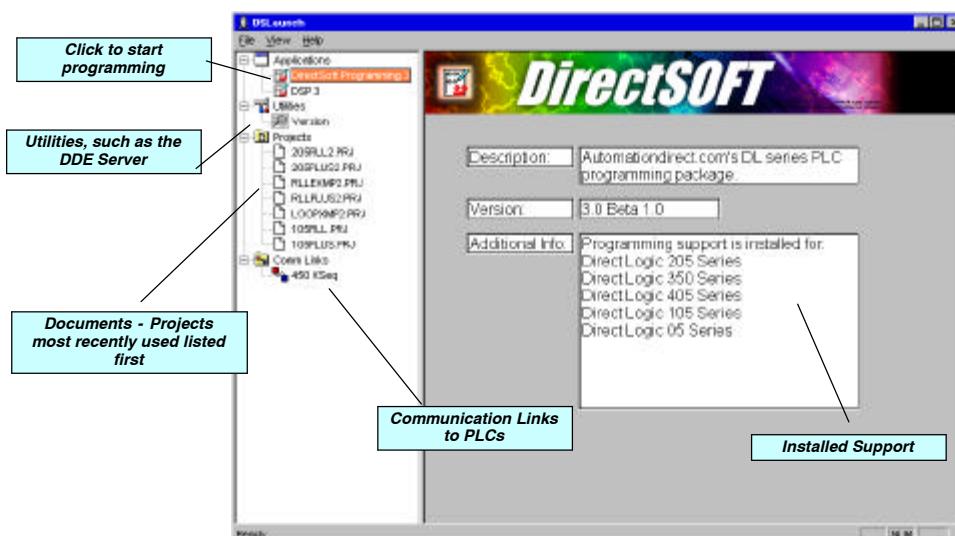
The ladder logic shown below is the same program for the DL05/105/DL205/DL305 and DL405 families. As you work through the **DirectSOFT** screens to enter this program in the pages that follow, the DL05/105/DL205/DL405 will be shown. If you have a DL305 family PLC, substitute the proper elements and memory locations. For example instead of entering **SP0**, you would enter **C374**.



**Step 1:
Enter the
Programming
Mode**

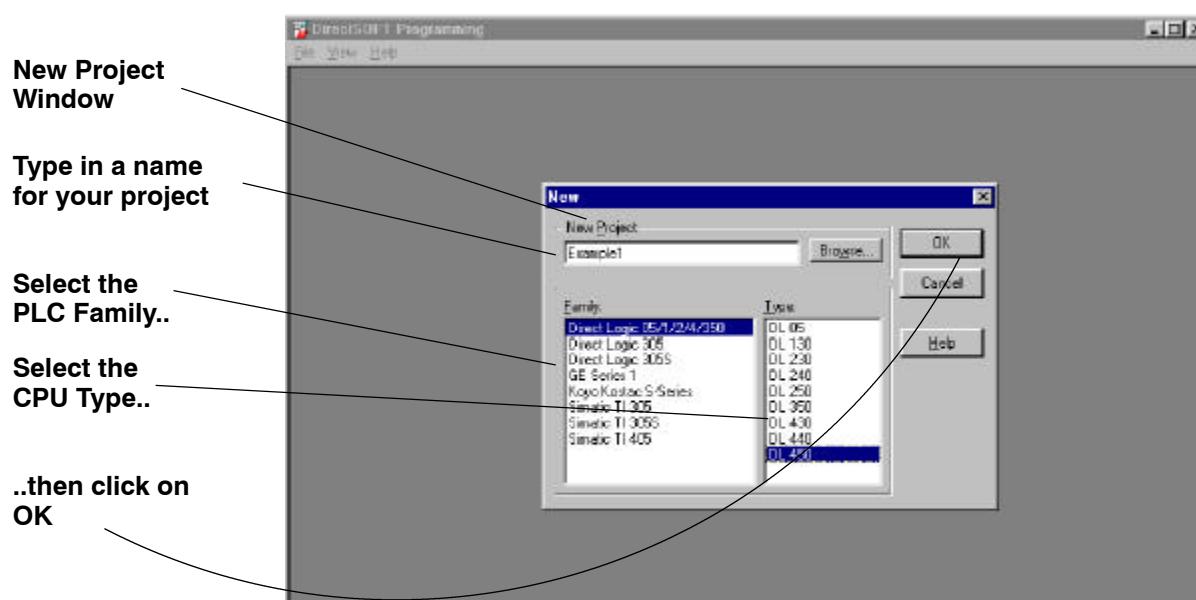
When you click on the **DirectSOFT32** Launch Window (rocket) icon the launch window similar to the one shown below appears. The Launch Window provides you with all of the information your current application has installed including what links it has detected, what PLC families are supported, what utilities are installed, and any existing projects.

Double click on the **DirectSOFT32 Programming** icon in the menu tree.

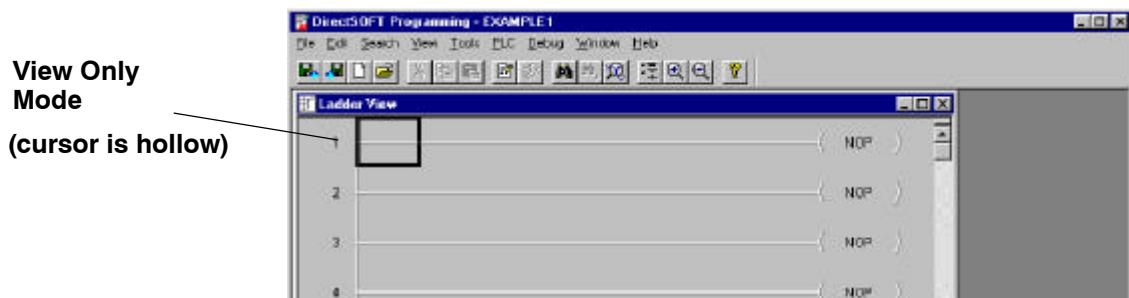


**Step 2:
Start a New Project**

You should now see the **New Project** window. You can name your project using any combination of 15 characters (including spaces). Use “**EXAMPLE1**” for this example. Move the selection bar to the **PLC Family** and **CPU Type**. For this example, use a PLC belonging to the DL05/105/DL205/DL405 families. Click on **OK** after you have made your Family and Type selections. If you have a DL305 type PLC, be sure and select it instead from the choices. Keep in mind the available mnemonics, processing rules and even the tool bar characteristics are tailored to the Family and Type selections that you make.



After clicking **OK** to enter your project name, PLC family and CPU type, you will see the skeleton ladder logic template. You are in the **View Only Mode** at this point. In this mode, the cursor is always hollow and all you can do is browse.

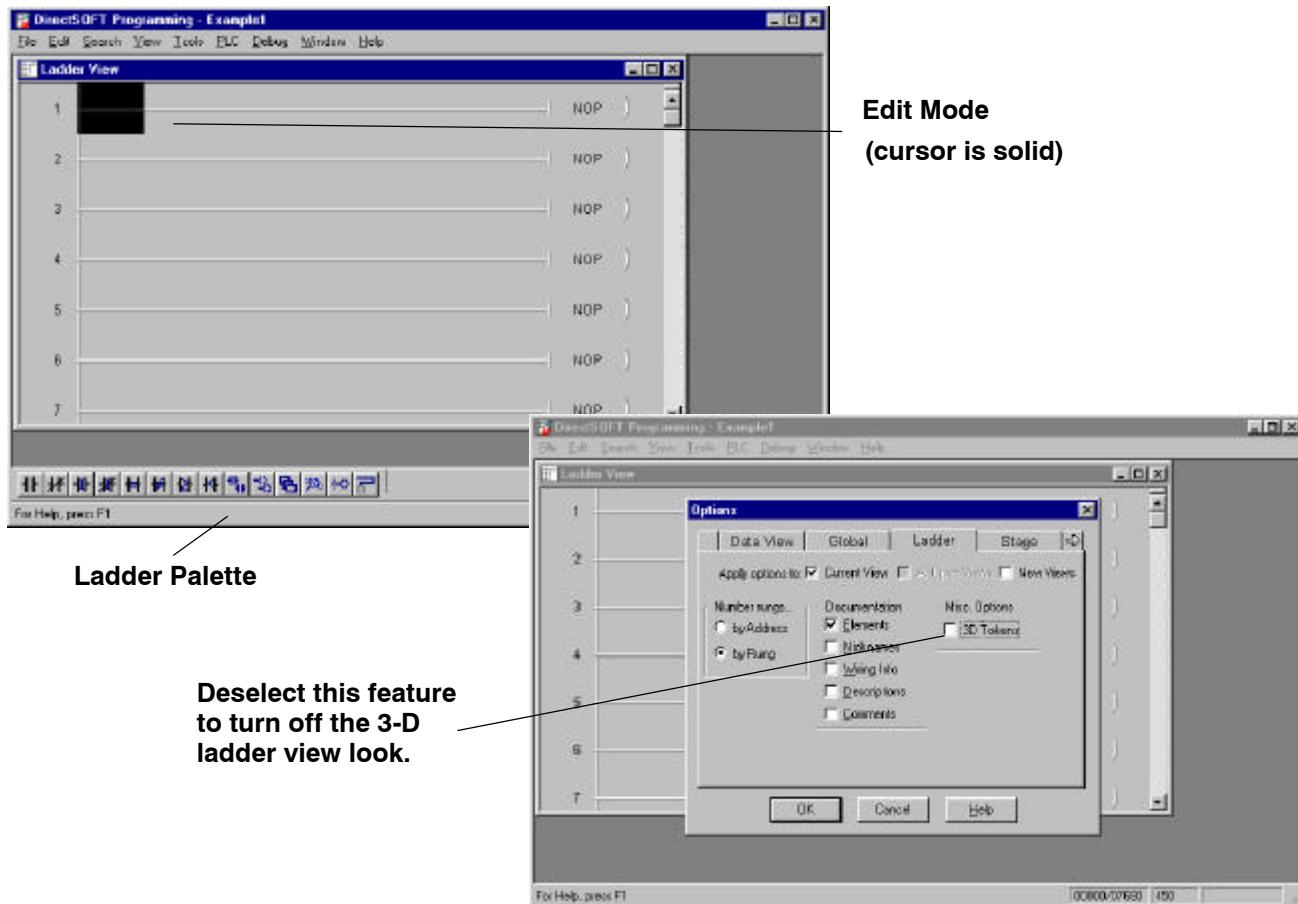


Step 3: Switch to the Edit Mode

You will want to use the **Edit Mode** for entering programs. This is accomplished by holding down the control key and simultaneously pressing the letter E key (**CTRL + E**). You could also click Edit on the top menu bar and then selected **Edit Mode**.

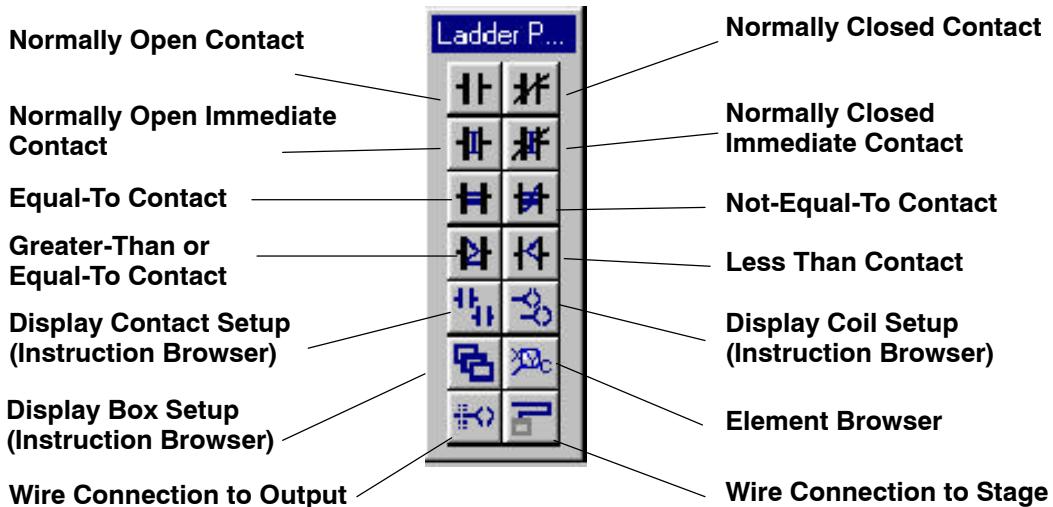
DirectSoft indicates the **Edit Mode** is active when the cursor becomes solid and a **Ladder Palette** appears in the lower left portion of the screen.

In order to facilitate clear printing reproduction in this manual, you will also at this time turn off the default 3-D Token for the ladder logic. You do this by clicking on View and then Options. This is followed by clicking on the box labeled **3-D Token**. This removes the X in the box. To exit and record your selections, click on **OK**.



Step 4: Using the Ladder Palette to Enter the First Element

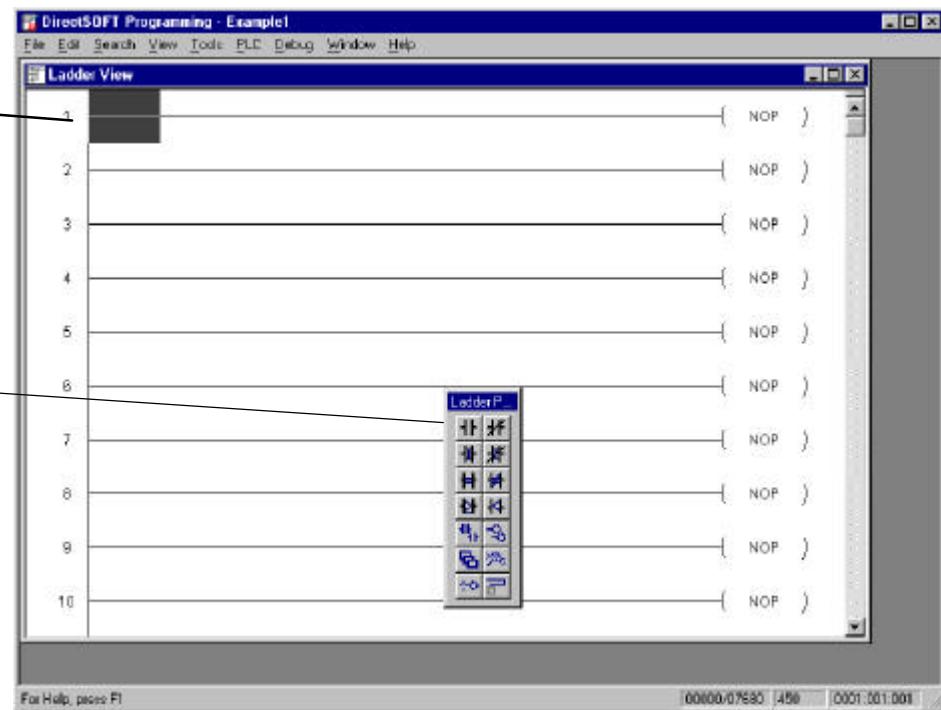
The **Ladder Palette** can be very helpful, especially in the beginning while learning to program in **DirectSOFT**. Later, you may prefer to use the faster **Hot Keys** instead of clicking on the tool buttons themselves. The Hot Keys appear when your cursor is on the element symbol icon. The expanded list is provided in the main **DirectSOFT** User Manual. *The Ladder Palette shown below may not be exactly like the one you have on your computer screen.* The Ladder Palette is floating and can be moved anywhere and to any layout that you prefer.



Use the **Ladder Palette** to enter the first instruction of the program. First move the cursor to the desired location of the first element. A normally open contact needs to be placed in the selected position to load the preset on the first scan. To do this, click on the button that shows the normally open symbol. This will open a small input window for setting up the contact.

Move cursor to where you want the instruction positioned.

Click here to open input window for Normally Open Contact

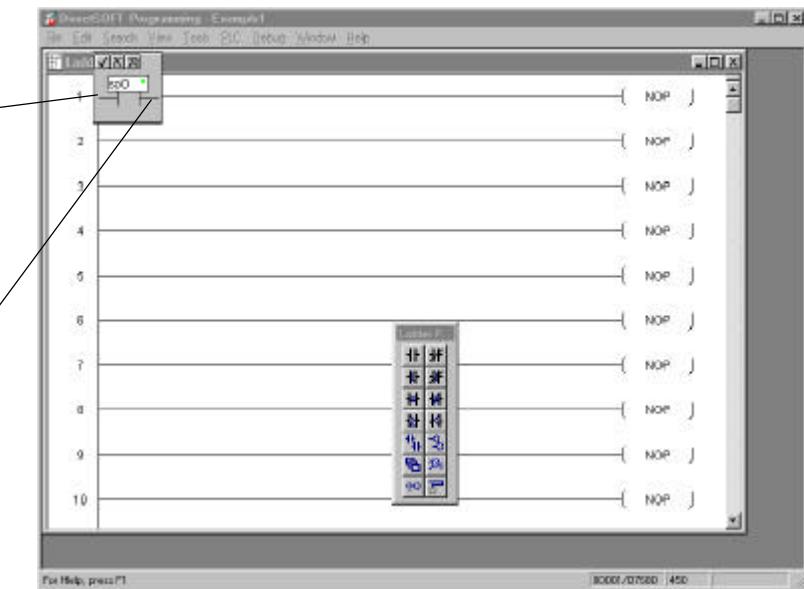


**Step 5:
Using the SP0
Relay in Our
Program.**

Use contact **SP0** as the first element to load a preset into the PLC memory. **SP0** is used because you only need this rung to execute once, i.e. the first scan. Notice the green/red indicator in the box. It will display the validation of each input. For example, if you typed the letter **O** instead of the digit **0**, the indicator would turn red and stay red until you correct your mistake.

Enter SP0 to have a contact that will close on the first scan only

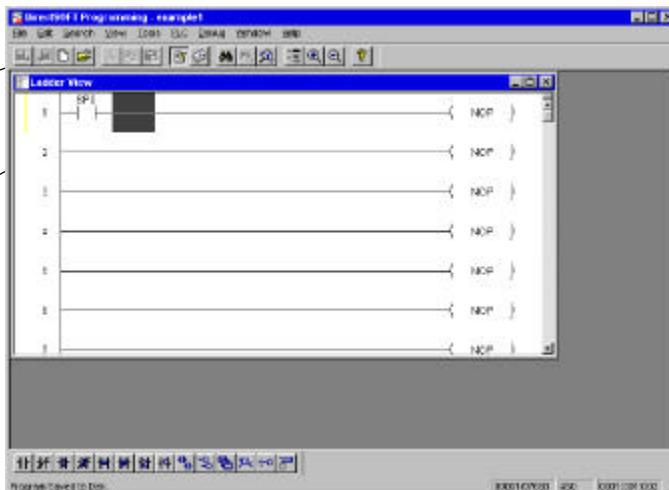
Notice the Error Indicator will glow green if you enter a valid contact



Click on the check mark (**✓**) in the upper part of the input box when you have finished typing in the element and have the green light. At that point, the instruction will be entered. Notice the yellow vertical bar that appears next to the rung. *Since this is not a color manual, you see a light gray vertical bar in the screen example shown below.* The yellow bar indicates you have entered an Instruction or instructions, but that you have not compiled the rung yet by selecting **Accept** from the **Edit** menu. Rungs that have already been accepted into compiled memory will have a green bar instead. Without being compiled, you will not see the icons for **Save to Disk** or **Save to PLC** enabled. This means in order to save your program anywhere you will have to **Accept** your editing first. For example, if you wanted to stop working with **DirectSOFT** right now, you would first want to accept all the edited rungs so that you could save the revised program to disk.

The Save to Disk icon is not available because you have not accepted your editing yet

Yellow color coded bar indicates the rung has not been accepted yet

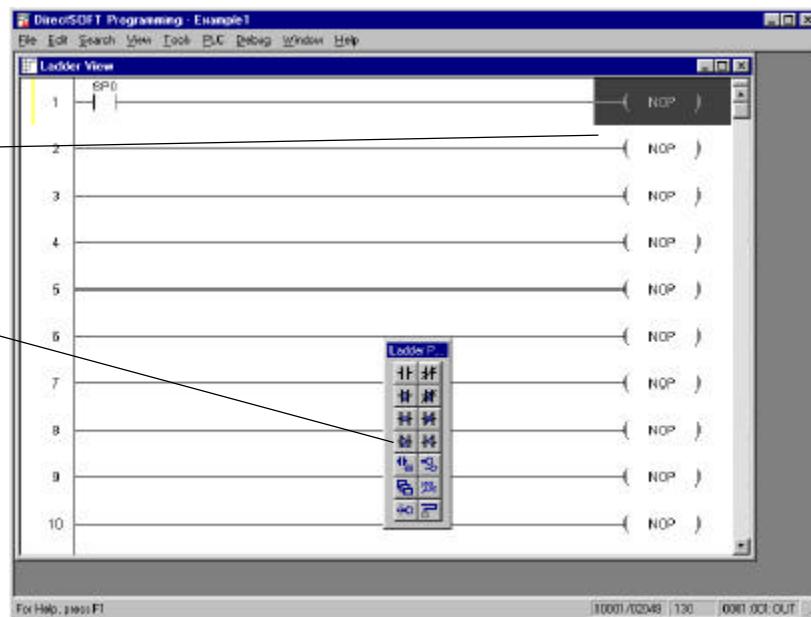


Step 6: Using the Box Browser

You are now ready for the output element on this rung. Move the cursor to the end of the rung to position where you want the element placed. Placing data in memory is a two step operation. First, load the data in the CPUs accumulator and then output it into memory. To accomplish this, *two output elements* will be placed on this rung. Start by entering the box instruction **Load Accumulator** that will load data into the accumulator. Once the cursor is positioned, click on the **Box** icon of the **Tool Palette**.

**Move the cursor
to the end of the
rung to position
the output
element..**

**..then select the
Box icon from
the tool palette**



Step 7: Scrolling the Box Class Window

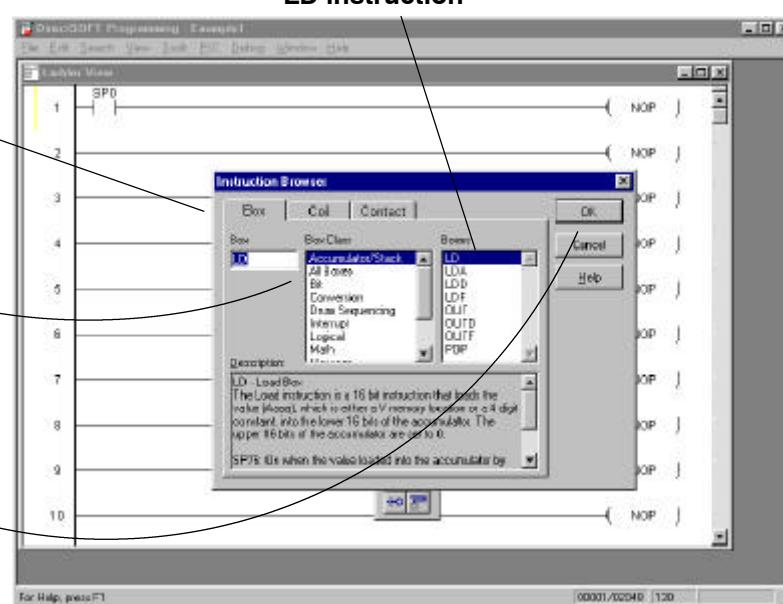
The box instruction to use for a DL05/DL105/DL205/DL405 is **LD**. This is found in the **Accumulator** class of the box instruction set shown in the **Box Setup** of the **Instruction Browser** that appears when you click on the **Display Box Setup** icon of the tool palette. With **Accumulator/Stack** and **LD** selected, click on **OK**.

**Use the Box Dialog to
select the instruction**

**Select the class in which
you find the instruction**

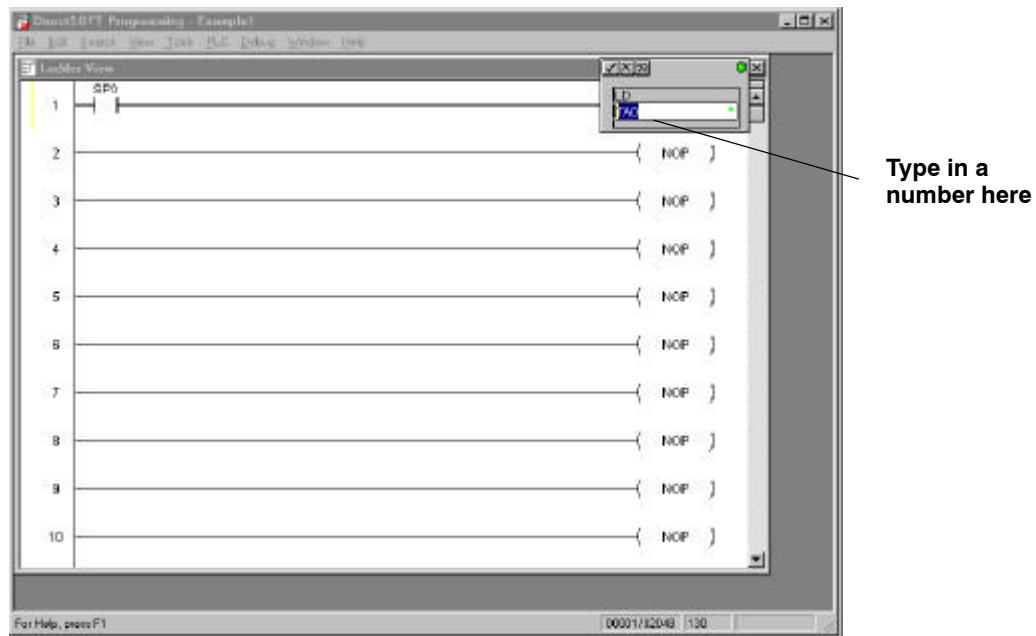
**Select OK when finished
with your selection**

LD instruction

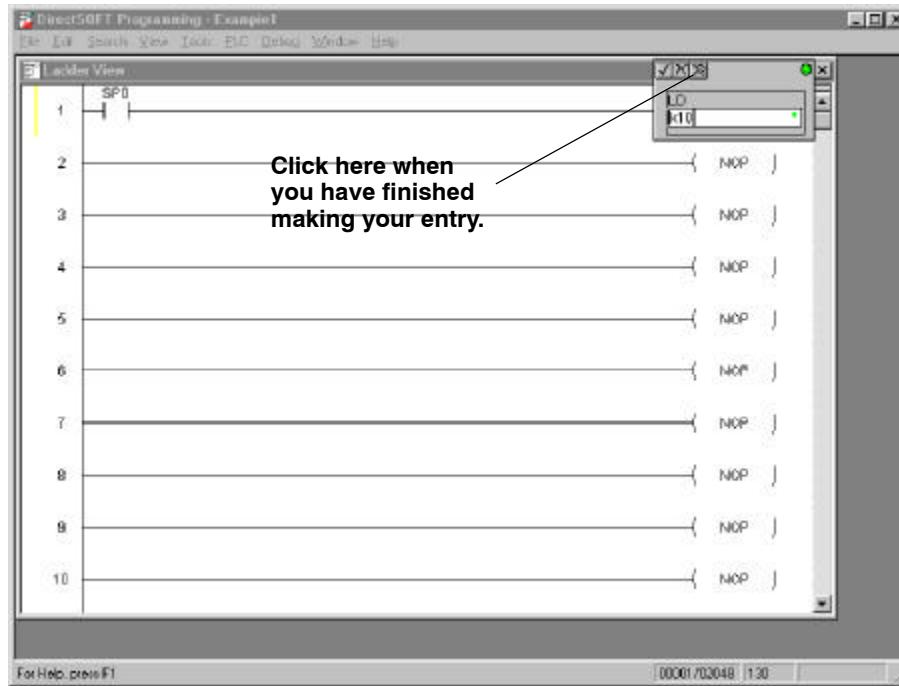


**Step 8:
Entering a Preset
in the Accumulator**

Now you see an input window that is very similar to the one used for entering the SP0 relay. It is waiting for you to type in the number to load in the Accumulator (recall from the program that you are using this first rung to enter the preset for the timer).



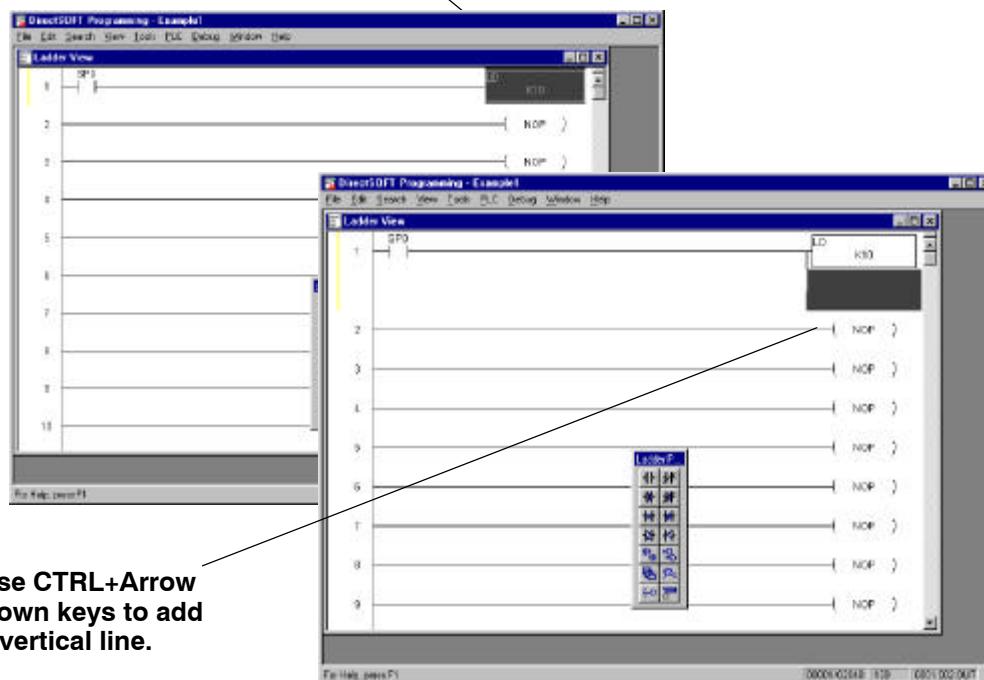
In this case use the number 10 as the preset. You will enter **K10** (where the **K** means **constant**). Again the green/red indicator inside the box will prompt you on whether or not you have made a valid entry. It glows green if it is correct. Click on (✓) when you have typed in **K10** and you “have the green light”.



**Step 9:
Connecting
Elements in
Parallel**

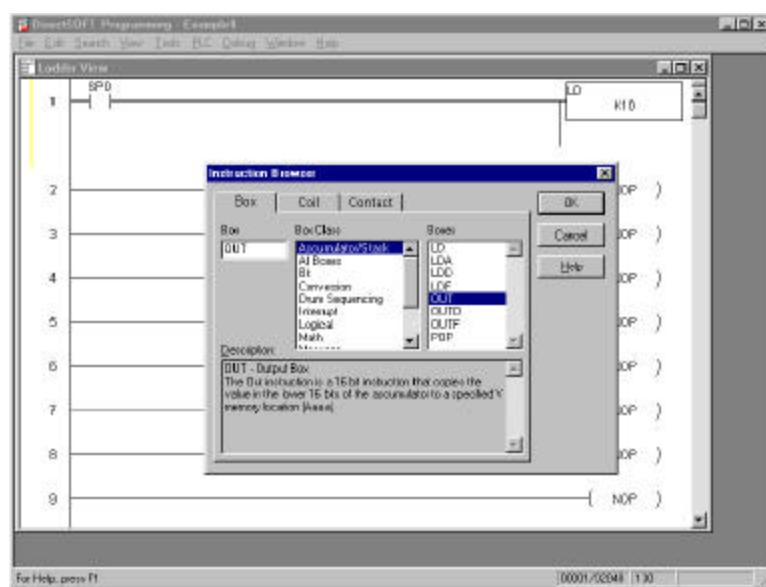
You are now ready to add a second box for this rung. It is going to be connected in parallel, therefore, you will need to add a vertical connecting line. With the highlight on the first box, hold down the control key and then press the down arrow (**CTRL + DOWN ARROW**). This draws the connecting line you need. You could have also performed the same thing from the upper menu bar by selecting: **Edit/Wire/Down**. However, the arrow keys are faster.

**Make sure the cursor is on the top element
before starting your vertical line.**



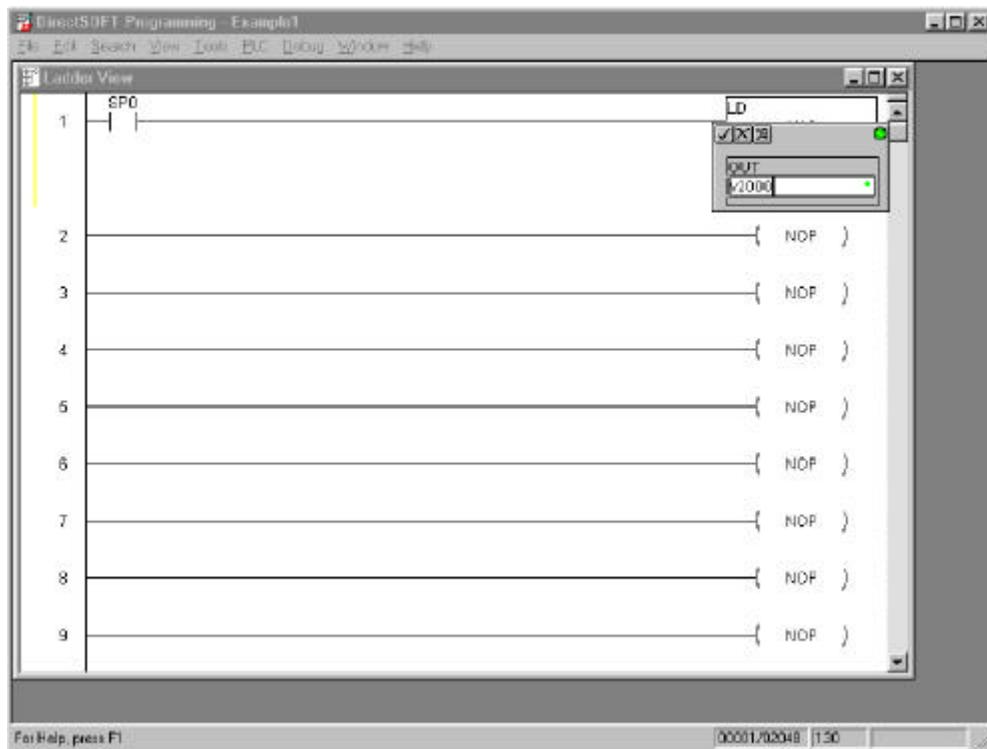
**Step 10:
Inserting an Output
Box**

With the cursor at the end of the line that you have created, select the Box icon from the tool palette again. This time, choose the **OUT** box from the **Box Tab** of the **Instruction Browser**. Select **OK** when you have highlighted the **OUT** function.

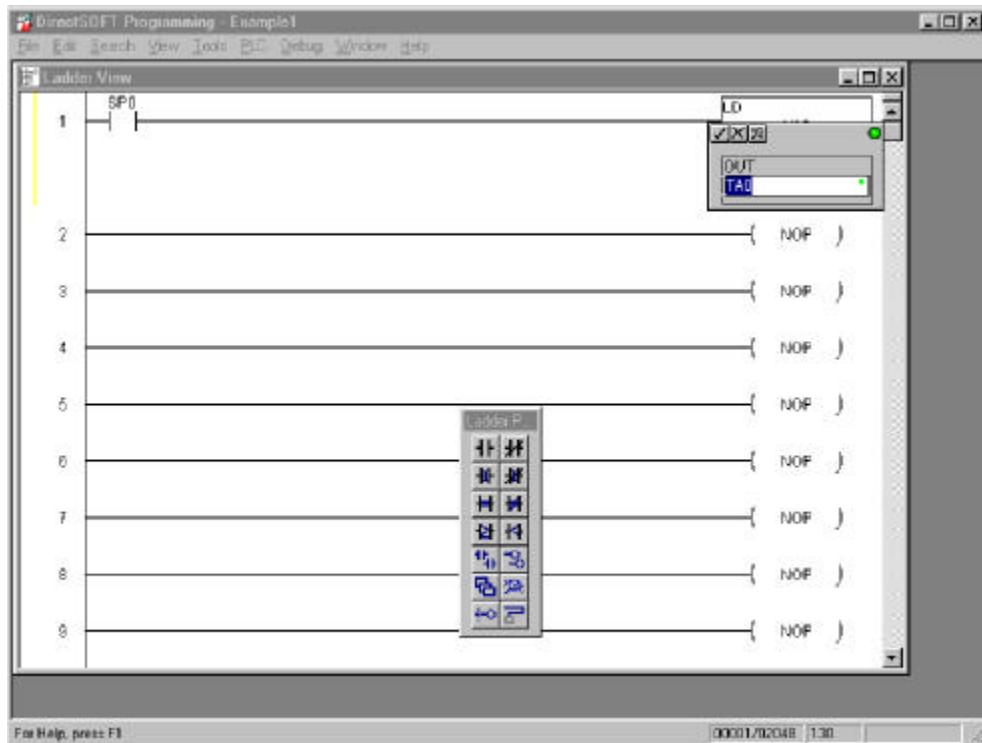


**Step 11:
Entering the
Address to Hold
the Preset**

Now you need to enter the address into which you are outputting the accumulator data. Use **V2000** in this example (**V** means “**variable**” memory).

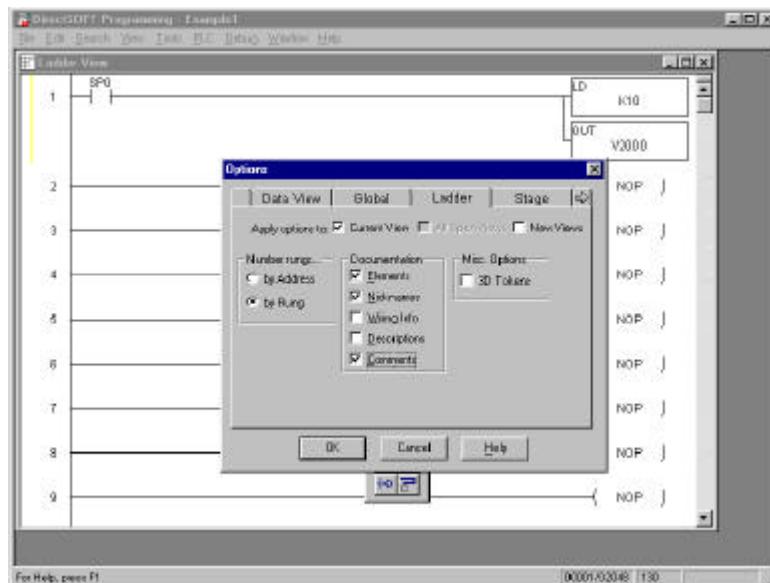


When you have entered the **V2000**, click on the check mark (✓). The new **OUT** function block should now be in place as shown below.



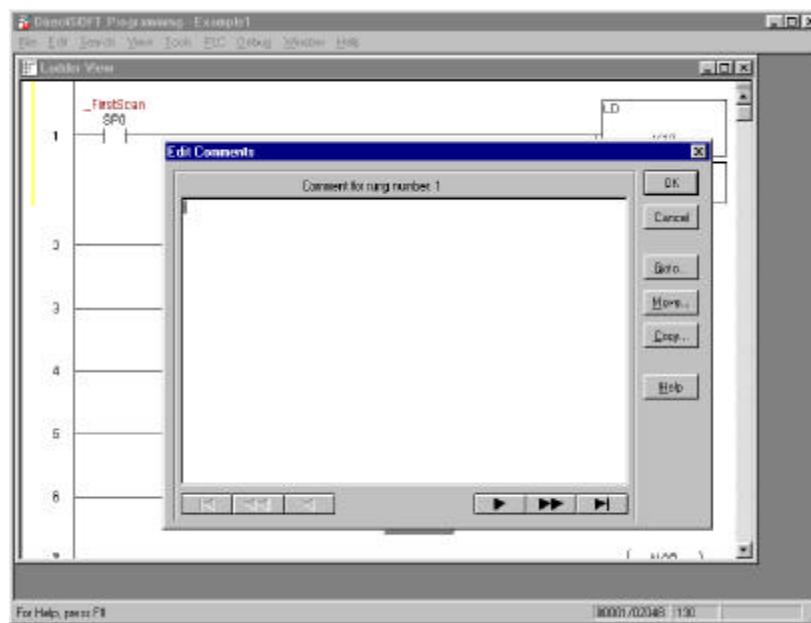
Step 12: Enabling the Documentation Options

You could stop with this particular rung at this point and go onto the next rung of our program. But in this example the idea is to make the program a little clearer to anyone who may look at it later. To accomplish this, you are now going to learn how to enter **comments** and **nicknames** for the program. Start by clicking on **View** from the menu bar at the top. Then select **Options**. An options menu appears (see below). Make sure nicknames and comments have been checked to ensure they will be displayed on the screen once entered. Select the boxes next to **Nicknames** and **Comments** to enable these options. Select **OK** when finished.

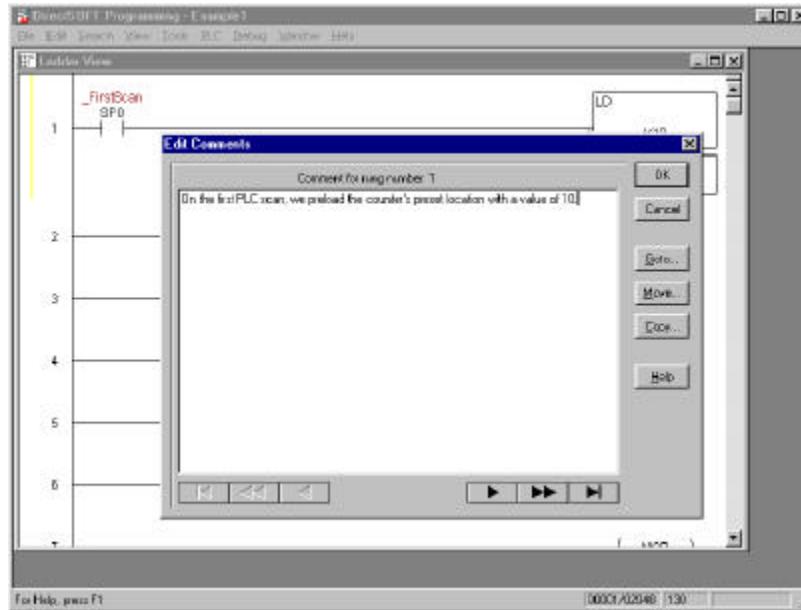


Step 13: Entering a Comment

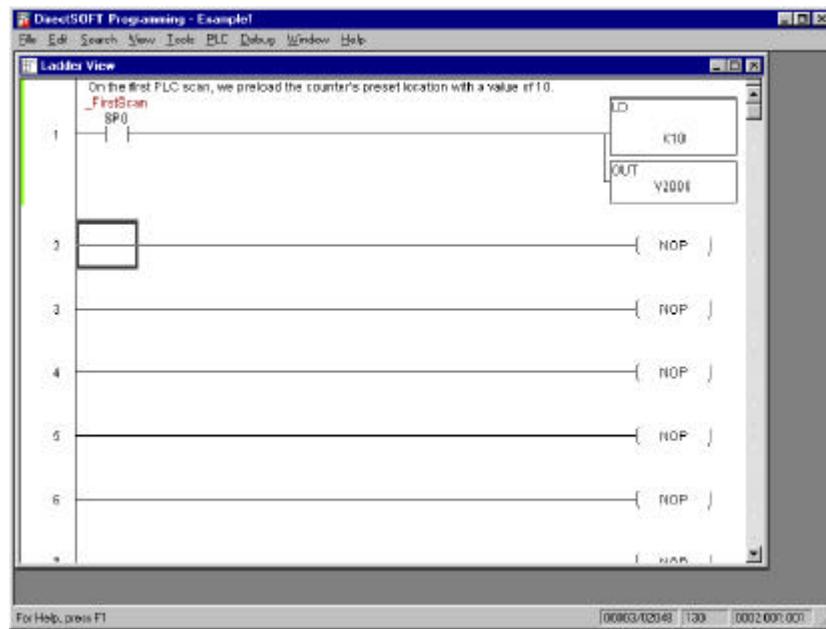
First insert a comment above the rung. To do this, you can pull down the menu from **Tools**, then select **Comment Editor** (Hotkey=CTRL + K). Either of these methods will display the **Edit Comments** dialog box shown below.



Now type in the comment you want and click on **OK**.

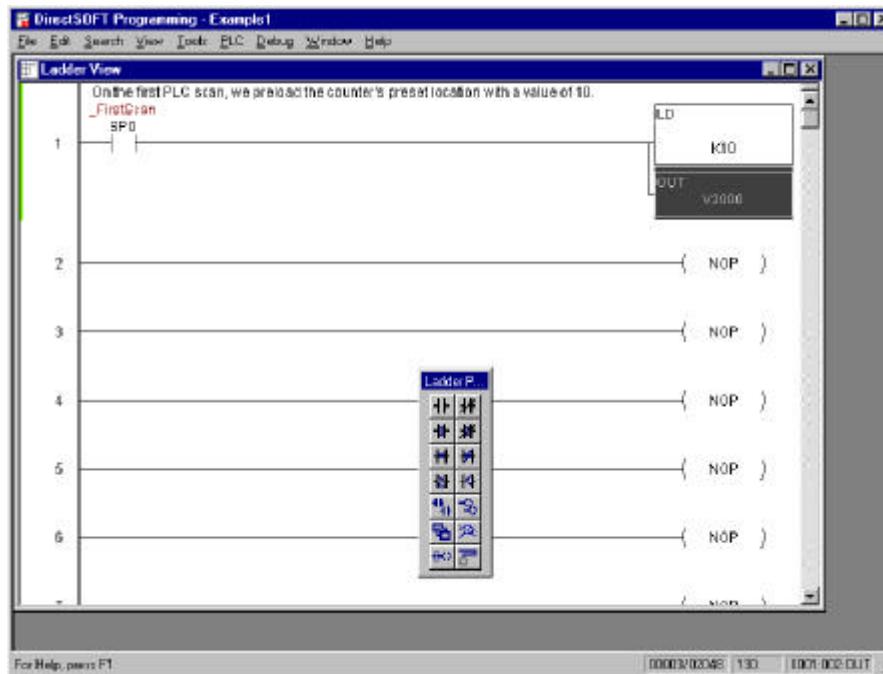


As a result, you end up with the comment shown above the rung like this:

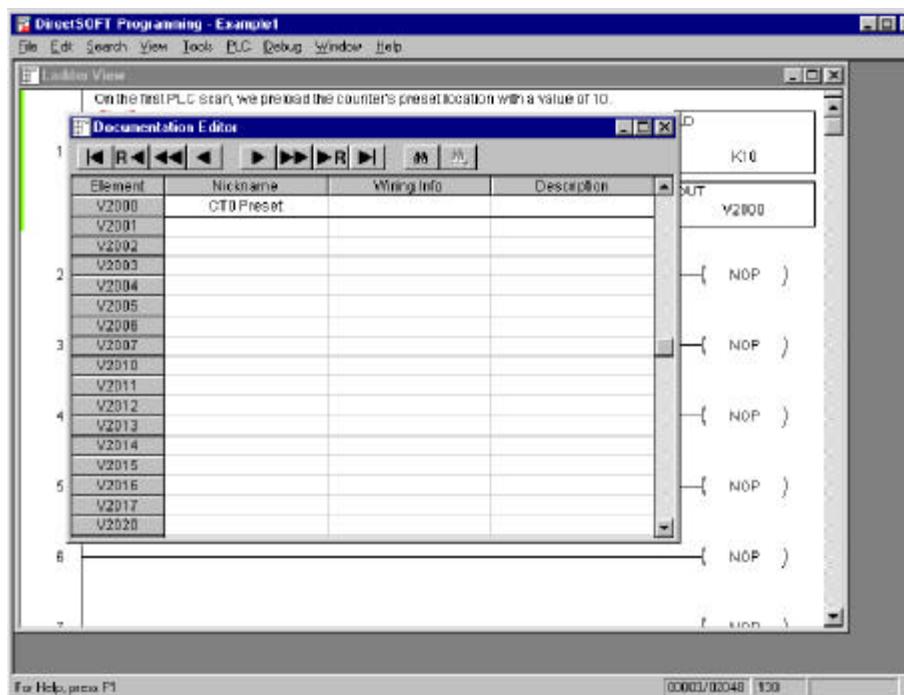


Step 14:
**Assigning a
 Nickname to the
 Preset Output**

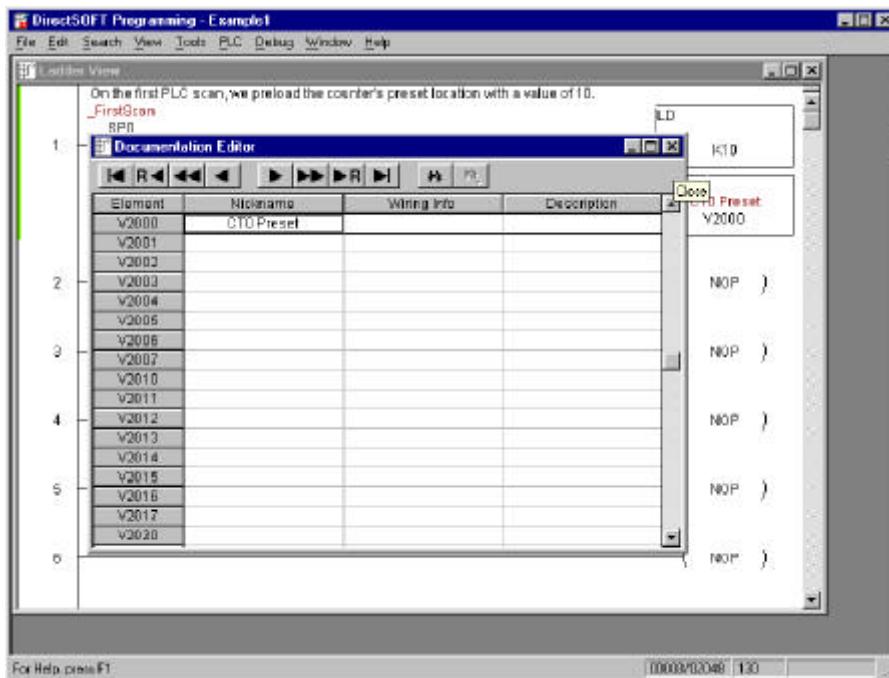
If you decide to assign a nickname to V2000, you will need to document that this address will hold the **preset** for the timeout counter **CT0**. Use a nickname that describes the function. Move the cursor to highlight the OUT box for the V2000. Select **Tools** from the upper menu bar. Next select the **Documentation Editor** or press **CTRL + D**.



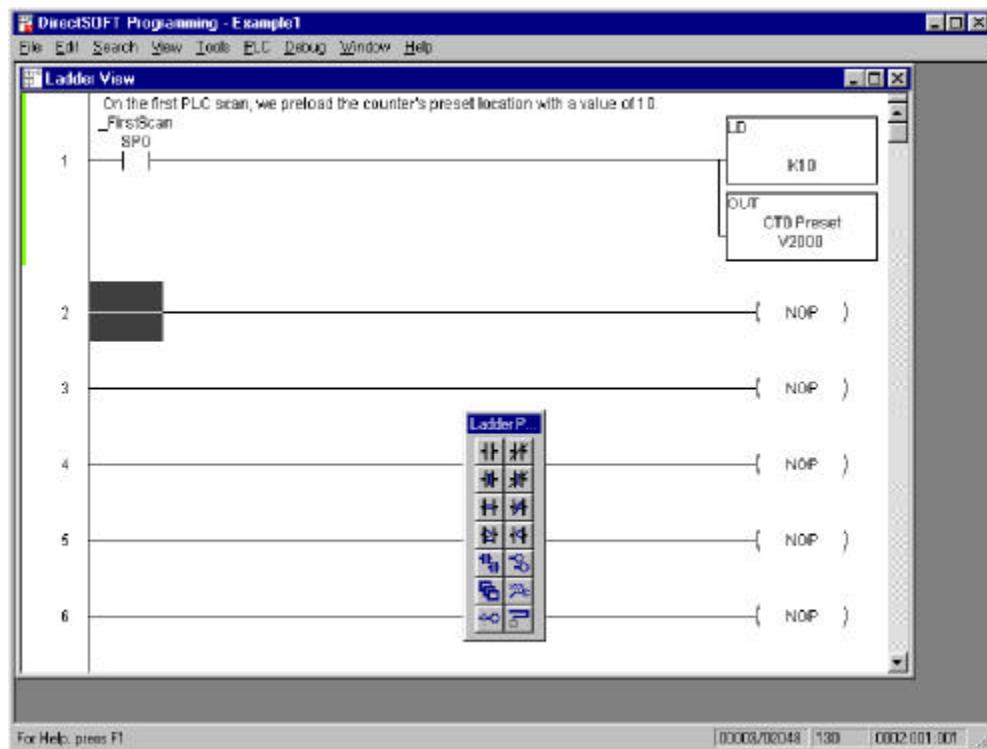
The dialog box shown below will appear. It will have **V2000** in the box labeled **Element**. Immediately to the right is the **Nickname** Box. Type in the nickname, in this case use the name **CT0 Preset**.



After typing in the nickname information, **Close the Documentation Editor** when you are finished.



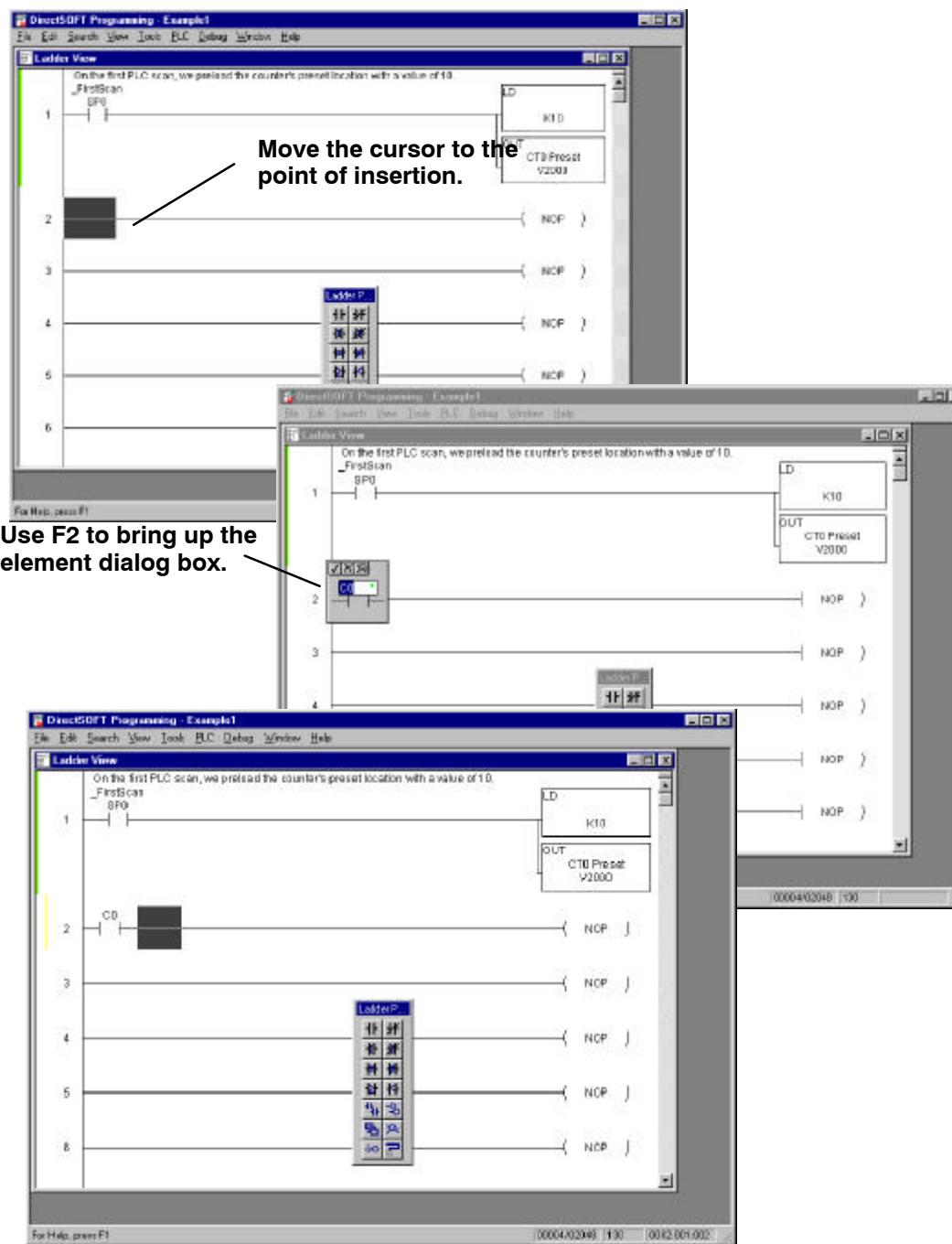
The nickname will appear inside the **OUT** box above **V2000**. You are now ready to create the second rung of ladder logic.



Step 15: Adding an Internal Relay to Start the Timer

In this next rung you will start a timer, have it timeout at a certain preset value, and then reset itself. Choose to use **C0** as the **start relay**; and of course, **T0** is the timer “done” bit for the **TMR T0** timer.

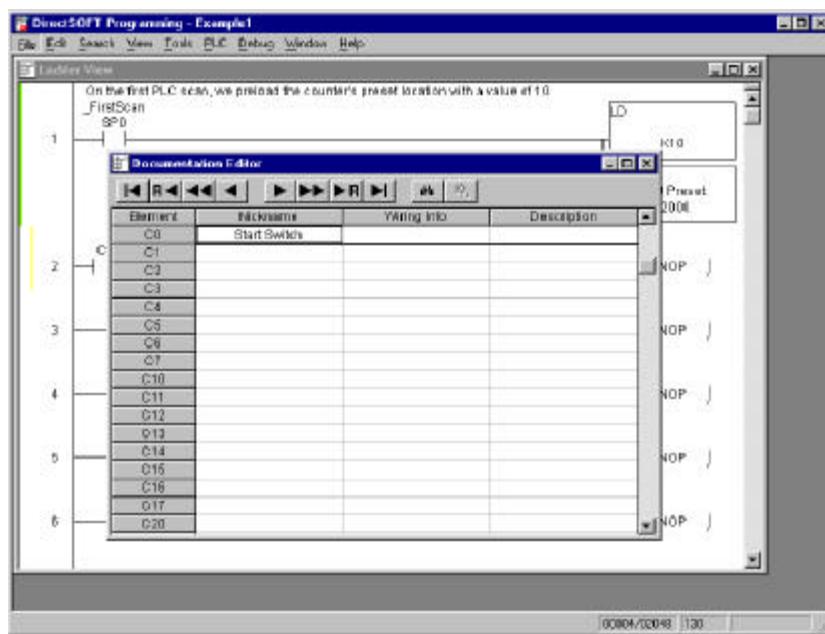
Insert the “start relay” **C0** first. Use a shortcut this time to create the normally open contact on the rung. You may recall on the first rung, the normally open contact icon was selected on the tool palette in order to create the contact. This time, move the cursor to the point where the contact is to be placed and press the **F2 key**. This opens the input box shown below and you can fill in the information as before. The shortcut keys are much faster than the icon selection method as you become more familiar with **DirectSOFT**.



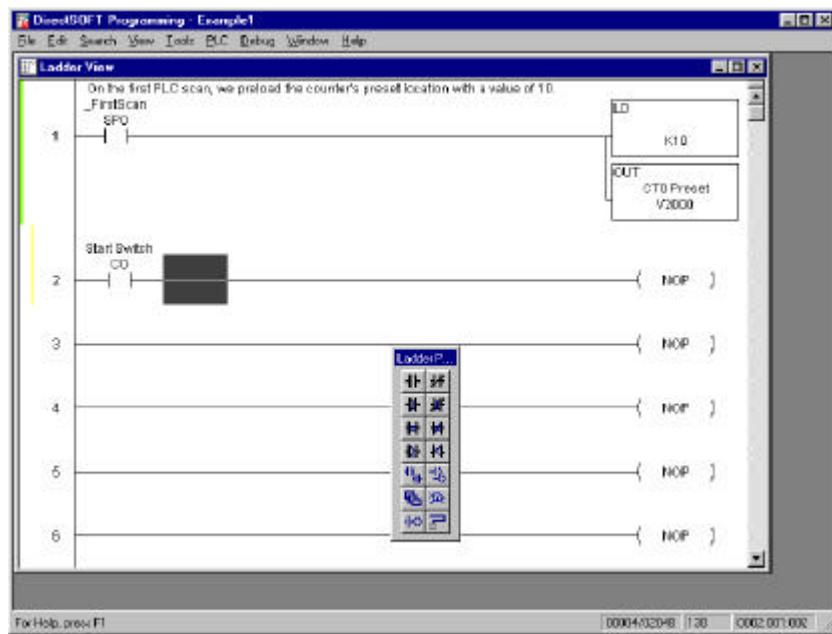
Step 16:
**Assigning a
 Nickname to the
 Start Contact**

Next, document that **C0** is the “start switch”. You could pull down the **Documentation Editor** from **Tools** on the horizontal menu bar to enter this information as a nickname. However, for learning purposes, use the hot key equivalent-**CTRL + D**. This will bring up the Documentation Editor without having to use the pull-down sub-menu of **Tools**.

If your cursor was on **C0** when you used the **CTRL + D** combination, the browser should have the **C0** in the **Element** column. Type **Start Switch** in the **Nickname** column.

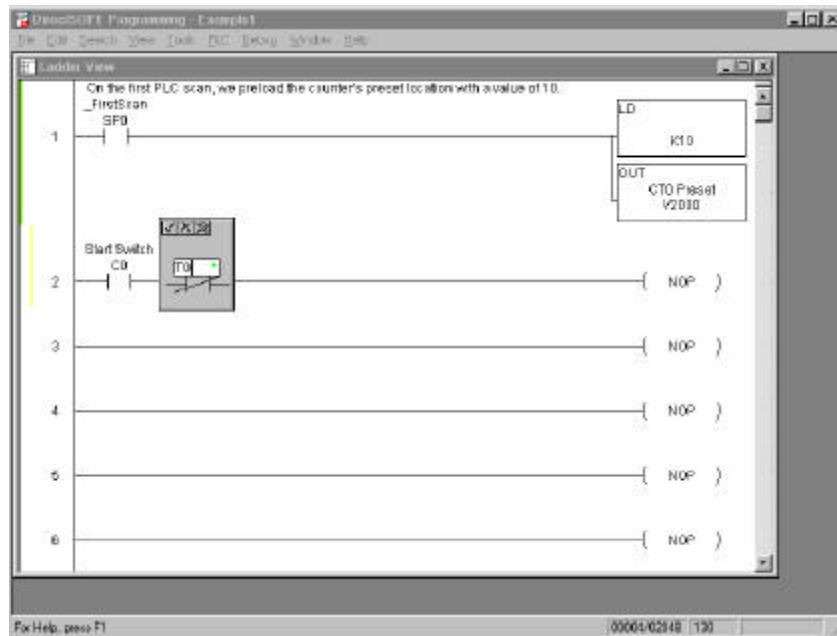


Close the **Documentation Editor**, returning to the rung. The nickname should be above the element **C0**.



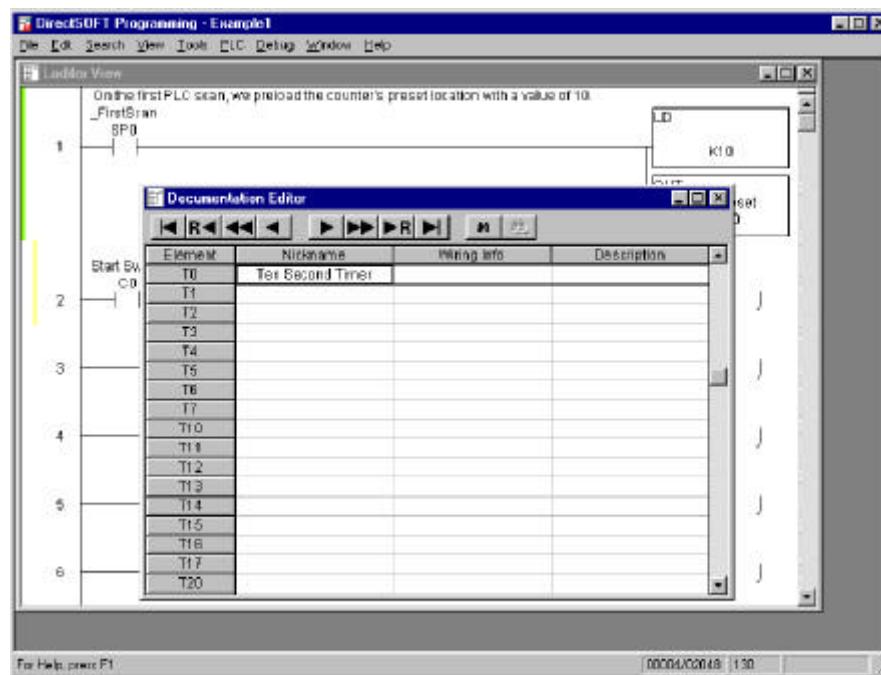
Step 17: Adding a Normally Closed Contact

Next, move the cursor further to the right on the rung to the point where you want to place the normally closed contact for the timer “done” bit (**T0**). Use shortcut key **F3** to bring up the dialog box for a normally closed contact. Type in **T0**. Finish by selecting the check mark.

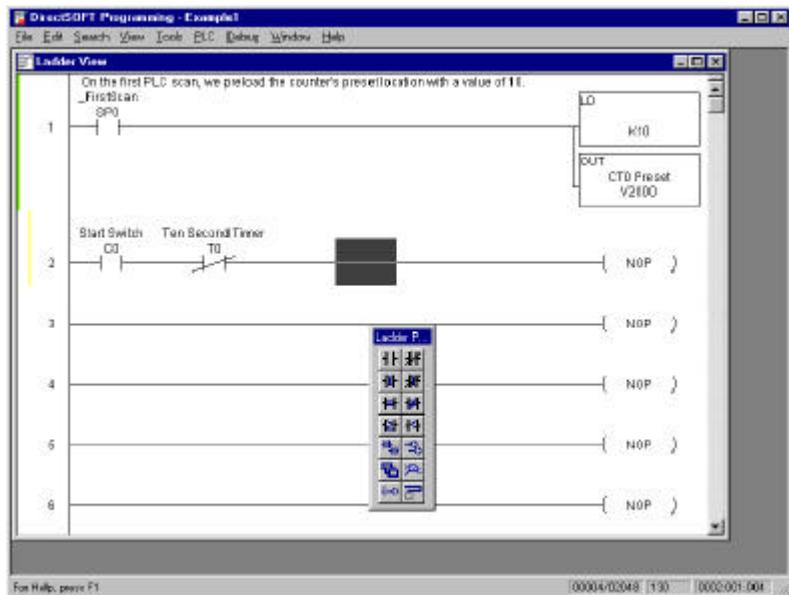


Bring up the **Documentation Editor** again by using the **CTRL + D** combination. Enter **Ten Second Timer** in the **Nickname** column. **Close** the Editor when finished.

Step 18: Adding a Nickname to the Normally Closed Contact

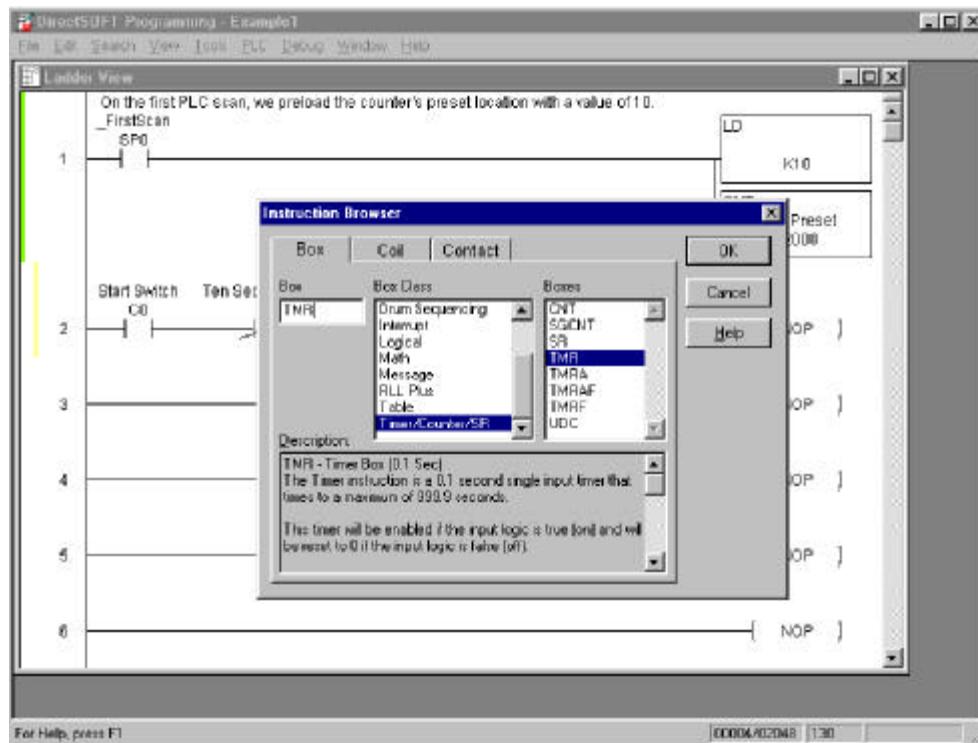


After the **Documentation Editor** closes, you will return to the rung where the words **Ten Second Timer** will be above the normally closed **T0** timeout relay.

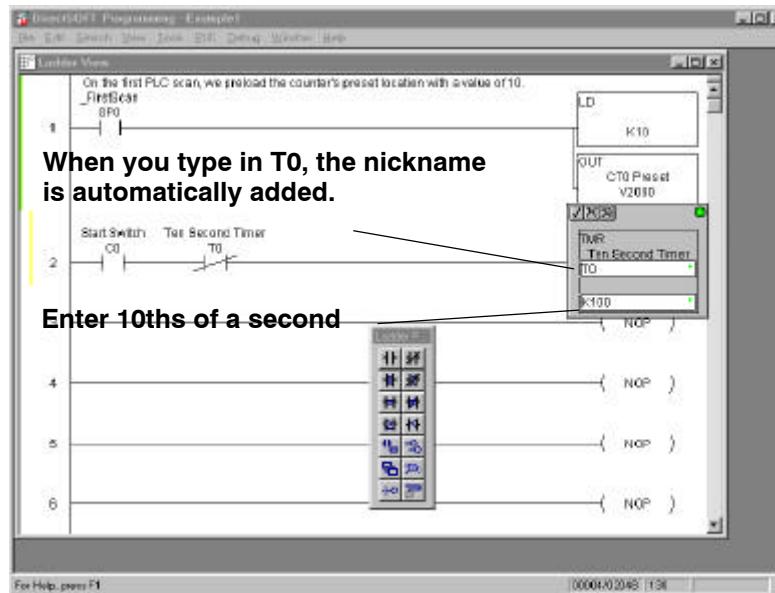


Step 19: Inserting the Timer Instruction

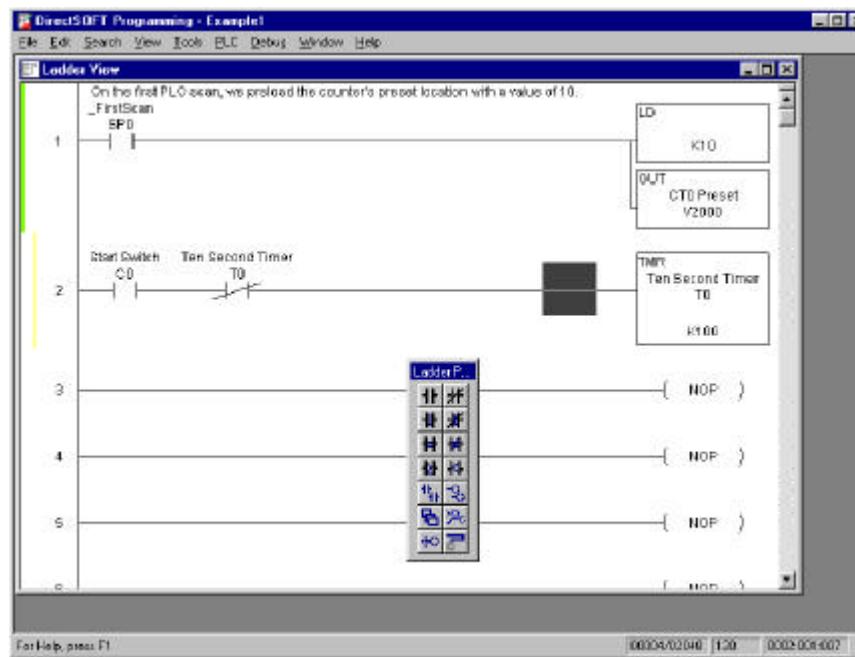
Move the cursor to the end of the rung to enter the timer **TMR T0**. Since a timer is a box command, you can use the shortcut key **F7** to bring up the box instruction dialog of the **Instruction Browser**. Once you have opened the dialog box shown below, move the cursor in the **Box Class** window to **Timer/Counter/SR**, move the cursor in the **Boxes** window to **TMR**, and click on **OK**.



The element box shown below will appear. This is the element input window for the timer. The first thing to do is allocate a particular timer. This example uses internal timer **TMR T0**, therefore type in **T0**. The indicator light of the box should turn green to tell you this is a valid entry. Notice also the nickname assigned earlier to the timeout relay **T0** appears above the first entry window. A second window also requires you fill in a preset value for the timer. Use the **Tab Key** or click with the mouse to move to the second field. The preset must be entered in tenths of a second. Therefore, **K100** would be one hundred tenths of a second (**100/10**), or **10 seconds**.

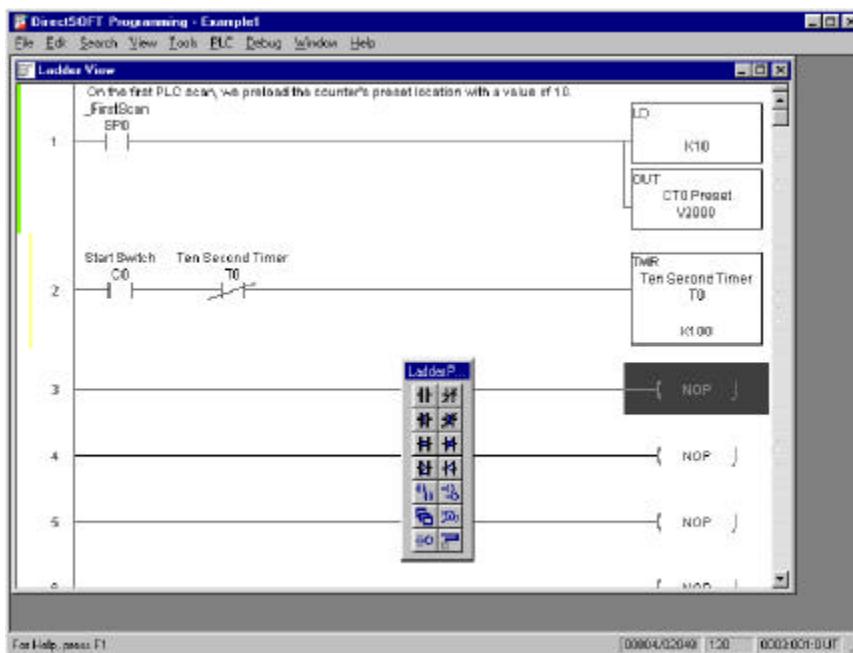


Click on the check mark to accept your entries. **DirectSoft32** returns you to the rung of ladder logic for further programming.

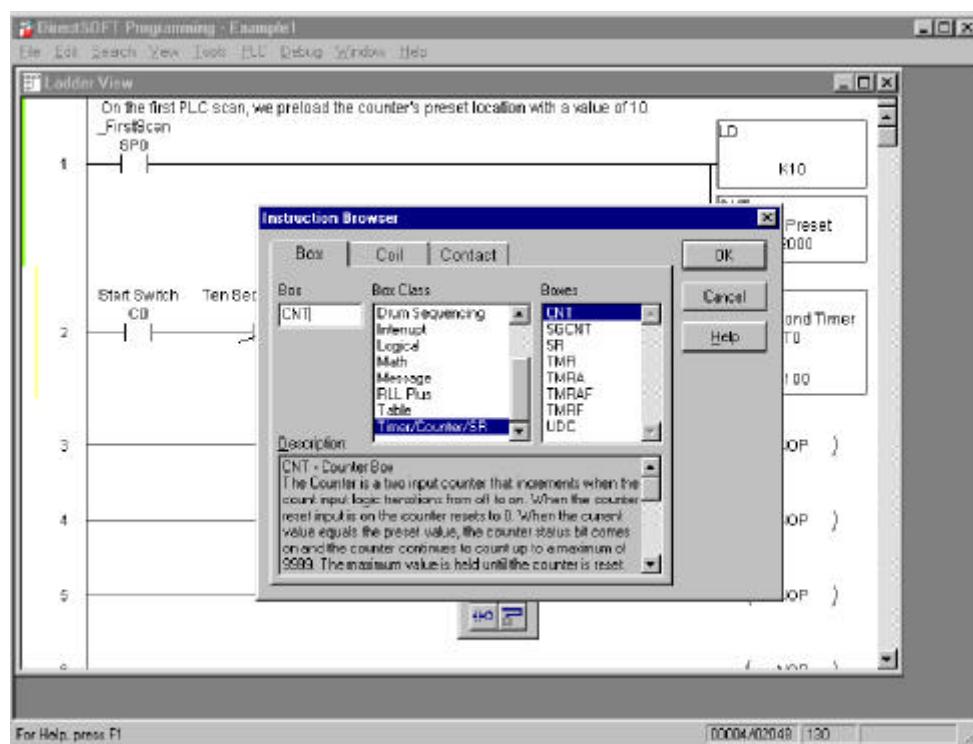


**Step 20:
Adding the
Counter**

You are now ready to start the third rung of the example program. You will be inserting the counter **CT0** (a preset was entered at V2000 with the first rung of logic). Move the cursor to the end of the third rung and press the **F7** key to open the **Box Tab** of the **Instruction Browser** again.

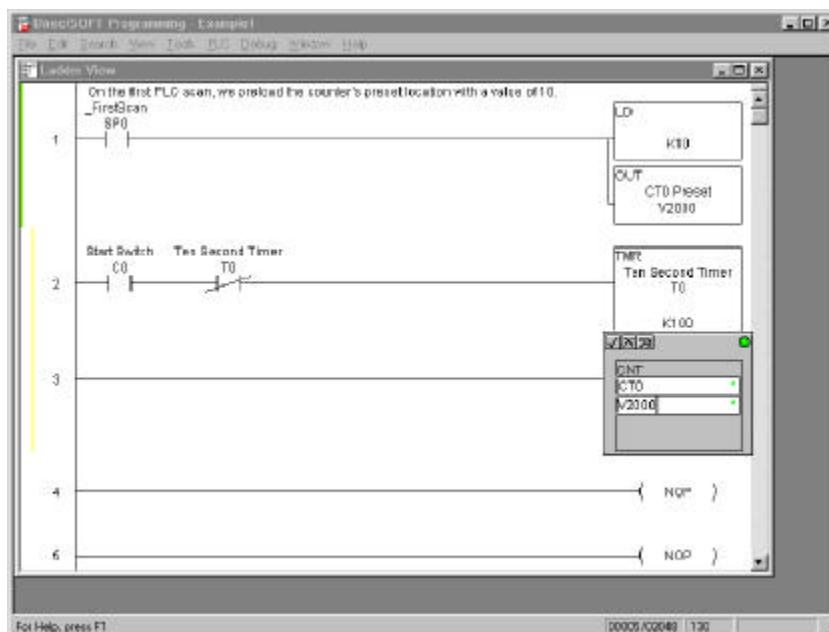


Select **Timer/Counter/SR** from the **Box Class**. Select **CNT** from the **Boxes** window and click **OK**.



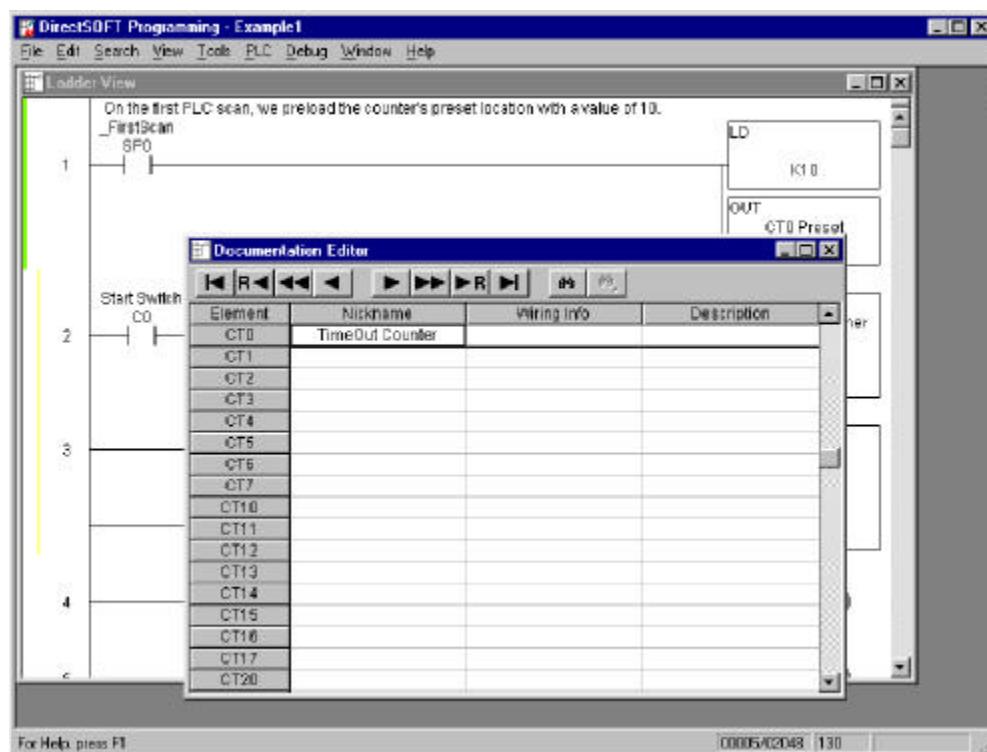
Step 21:
Entering the
Counter Data

An element window will appear. Enter **CT0** for the counter and **V2000** as the address holding the **preset** data for the counter. Select the check mark when you are finished making the entry.



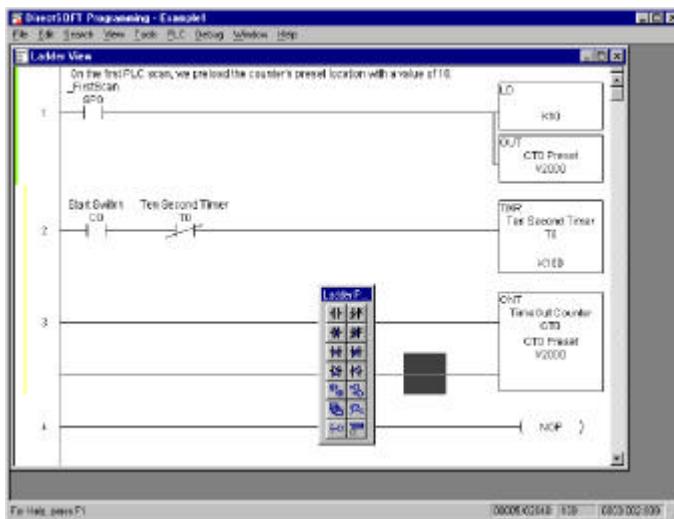
Step 22:
Assigning a
Nickname to the
Counter

Enter the **Nickname** (TimeOut Counter), using the same procedure previously described. Use the key combination **CTRL + D** to bring up the browser.



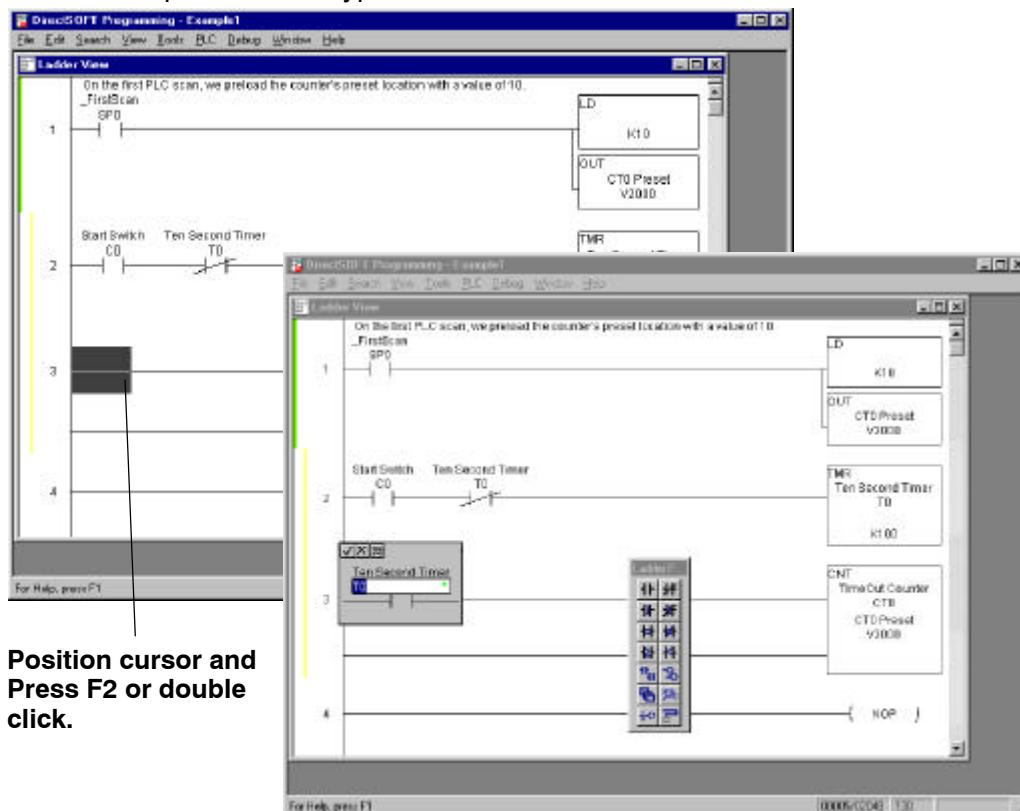
**Step 23:
Making the
Counter
Self-Resetting**

Close the Documentation Editor and return to the rung. You will see the new Nickname, as well as the Nickname (**CT0 Preset**) you had given earlier to the preset memory location V2000.



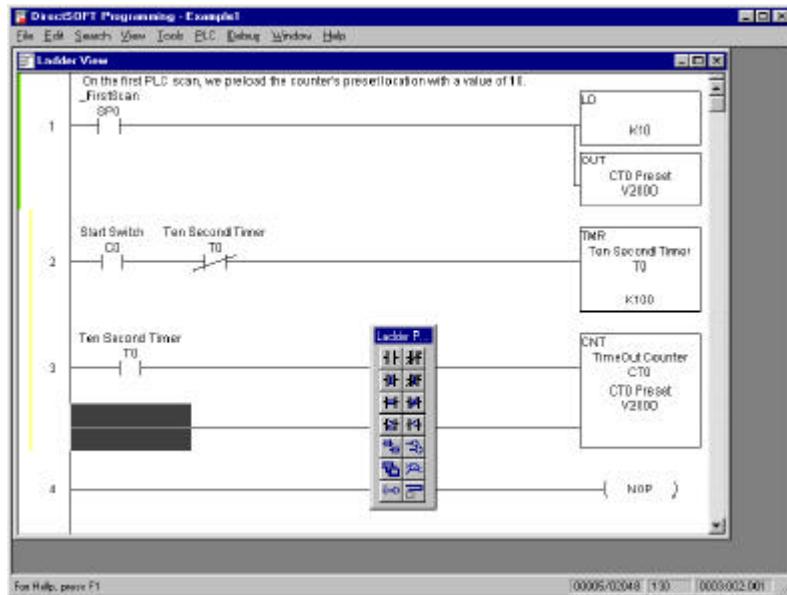
Notice the **CNT** box has two inputs--**count enable** and **reset**. To count the number of times the “done” bit turns ON, the contact for the timer “done” bit (**T0**) on the **count enable** rung needs to be inserted.

With the cursor in the position shown below, enter contact **T0**. Press **F2** to call up the contacts input window. Type in **T0**. Select the check mark when finished.

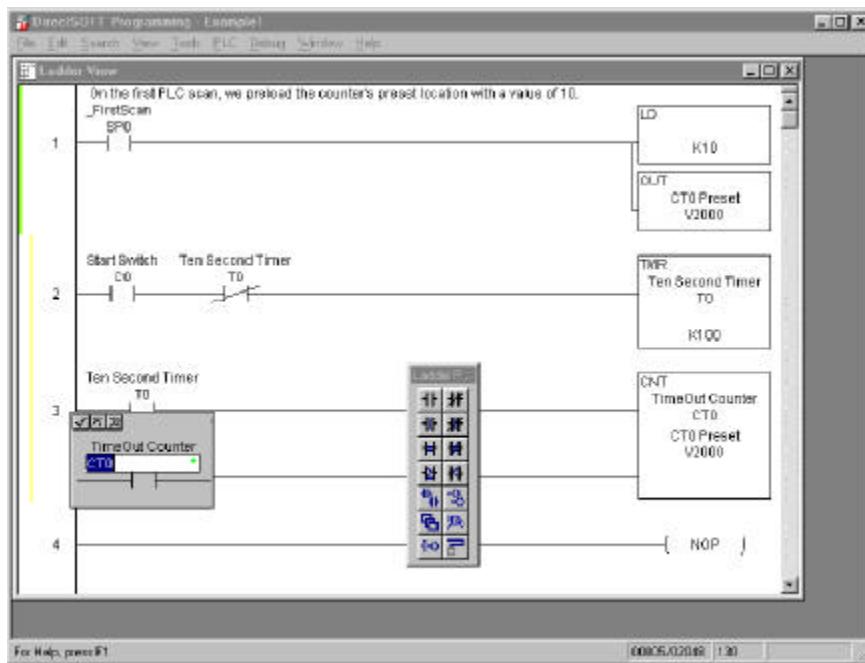


Notice the **Nickname** assigned previously for **T0 (Ten Second Timer)** automatically appears to the first rung of the counter.

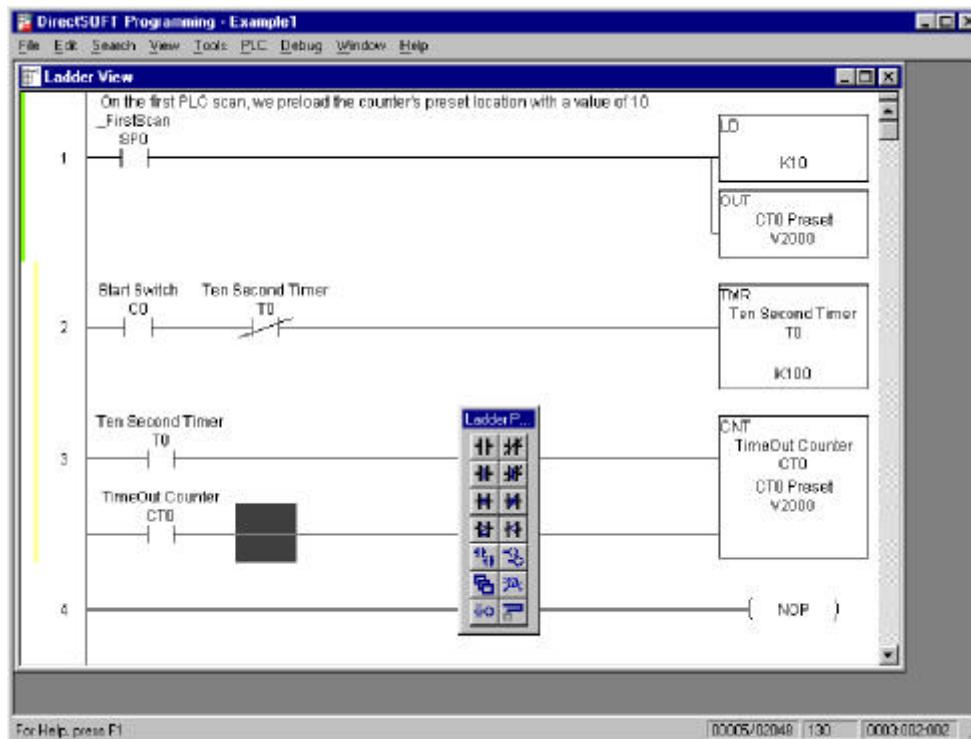
With the count enable rung of the counter completed, move the cursor down to the second rung of the counter to enter the reset logic.



At this point, the reset contact using the counter “done” bit (**CT0**) will be entered so when the counter reaches its preset, it will automatically reset itself to zero. Again, press F2 to bring up the contacts input window. This time type in **CT0**.

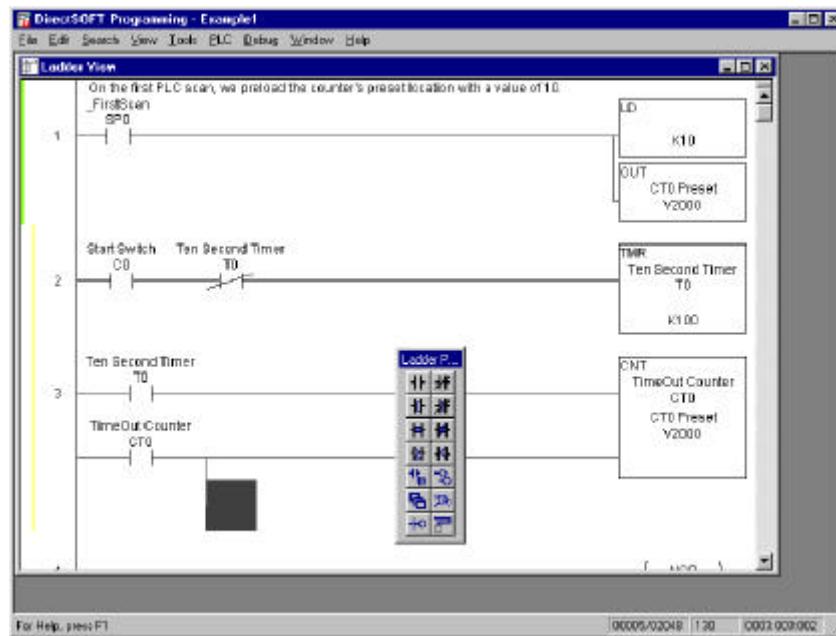


Click on the check mark in the dialog box to return to the rung. Notice the **Nickname for CT0 (TimeOut Counter)** is automatically placed above **CT0**.

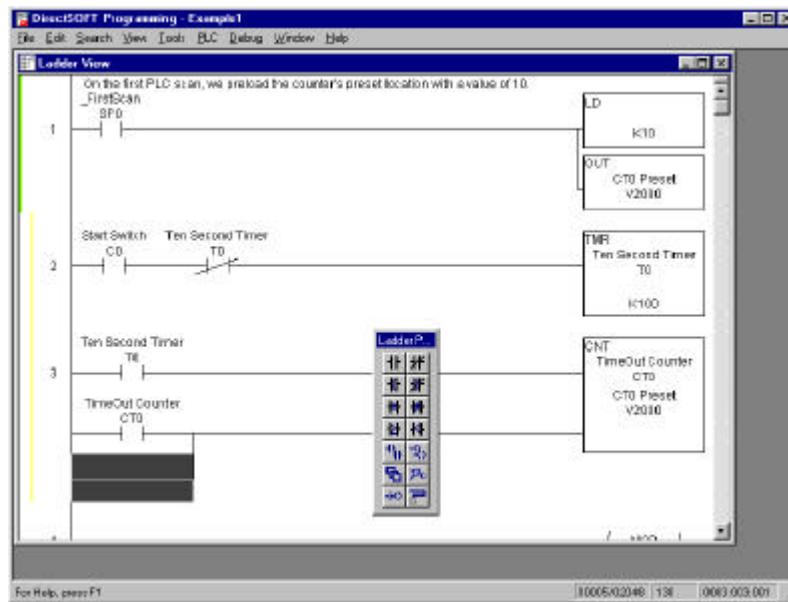


Step 24: Adding a Coil for Resetting on the First Scan

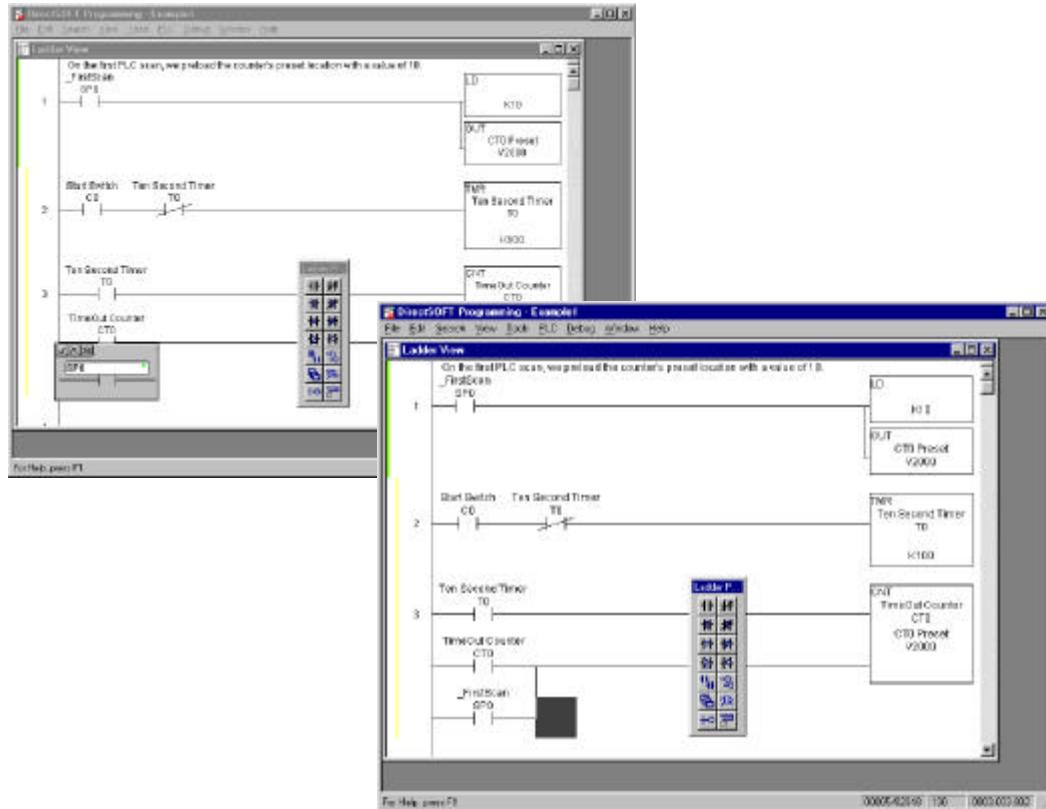
You will want to reset the counter during the first scan. The counter will reset on the first scan by placing special relay **SP0** in parallel with the reset contact (**CT0**). To place an instruction in parallel with another, first position the cursor to the right of the first instruction and use **CTRL + DOWN ARROW** to place a vertical connecting segment extending downward.



Press the **F2** key to bring up the normally open contact input window.

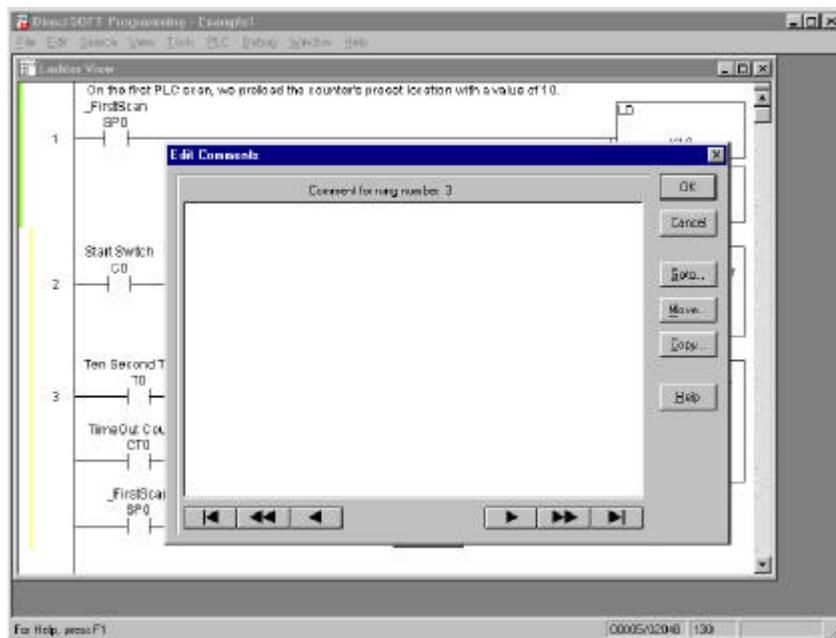


The special contact **SP0** turns **ON** for the first scan will be entered next. You do not have to place a nickname above SP0. The software automatically places **_FirstScan** above it. This is a “system-defined” nickname. You will find a list of special contacts and nicknames in an appendix near the end of your PLC user manual.

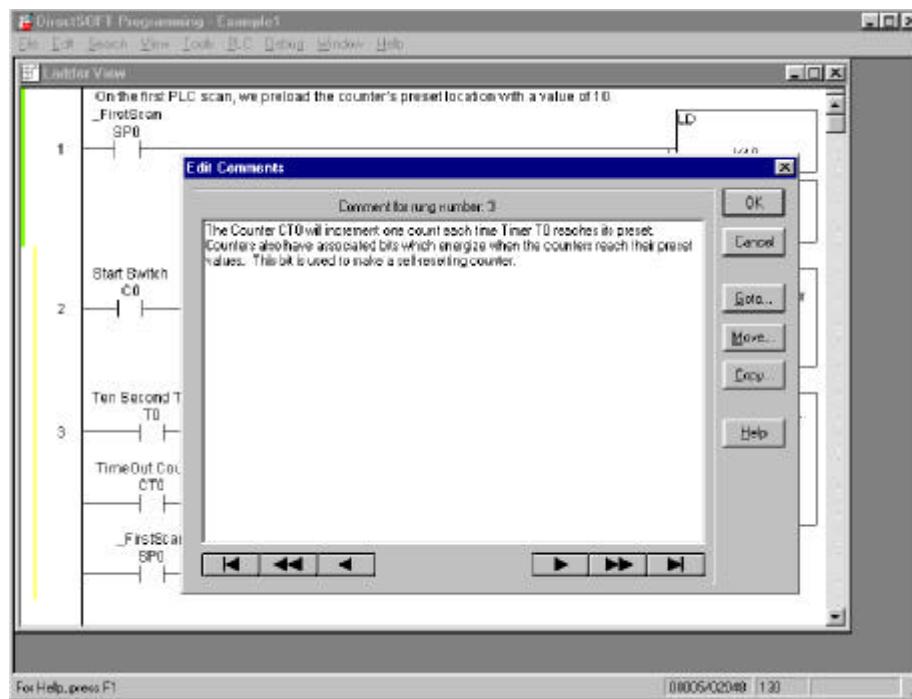


Step 25:
Documenting the
Function of the
Counter

In this example, refer to the top rung of the counter and add a comment about the function of **CT0**. You will use the same **Edit Comments** dialog window as in the first rung. This is opened by pressing the shortcut key combination **CTRL + K** or by double clicking anywhere above the rung to which you are adding comments.



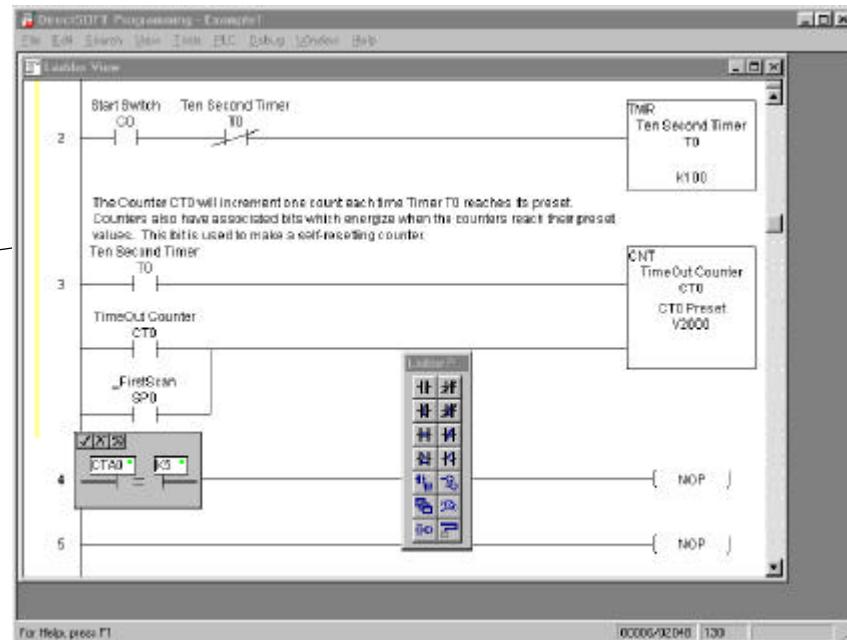
Now type in your comments for this part of the ladder logic. When finished, select **OK**.



**Step 26:
Adding a
Comparative
Boolean
Instruction**

You are now ready to start a new rung. The next rung of logic will turn ON an output when the counter reaches a count of 5. Use the Tool palette to open the **Equal To** (Comparative Boolean) dialog. Type in **CTA0**, which is the **DirectSOFT** name for the accumulated value of counter **CT0**. Tab to the right side of the input window to enter **K5**. Select the check mark when you have entered the constant value, **K5**.

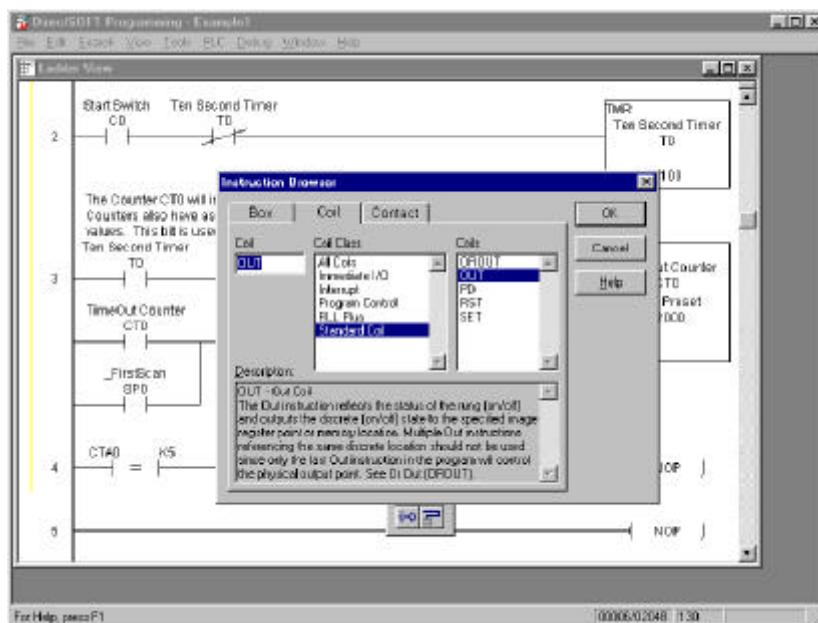
Click on this icon to open “equal to” dialog.



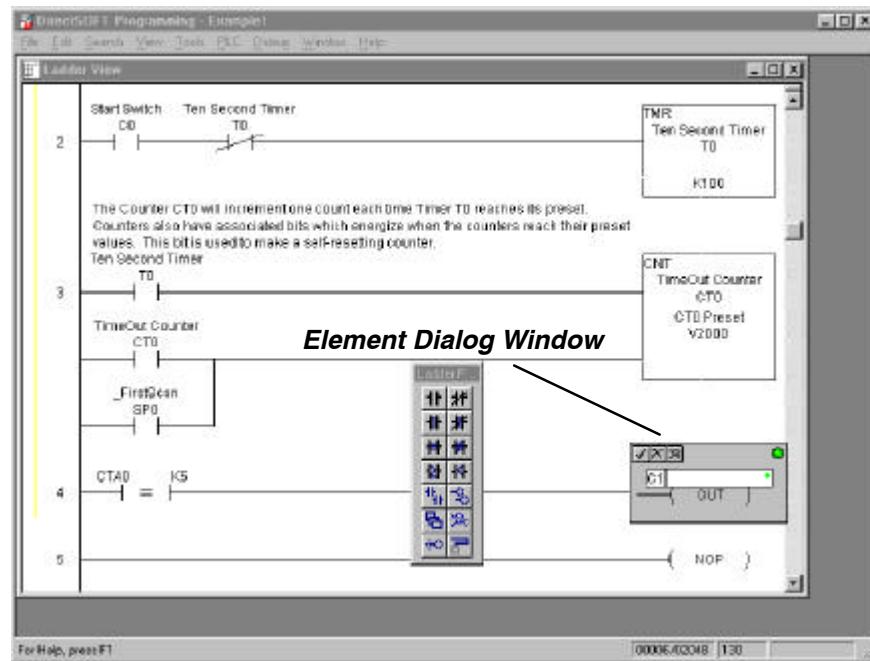
**Step 27:
Adding a
Conditional Output**

In this example, you will use **C1** as a test output coil. You will be able to see if **C1** turns ON by viewing the screen during the running of this program. As an output turns from OFF to ON, there is a color change on the screen for that particular element.

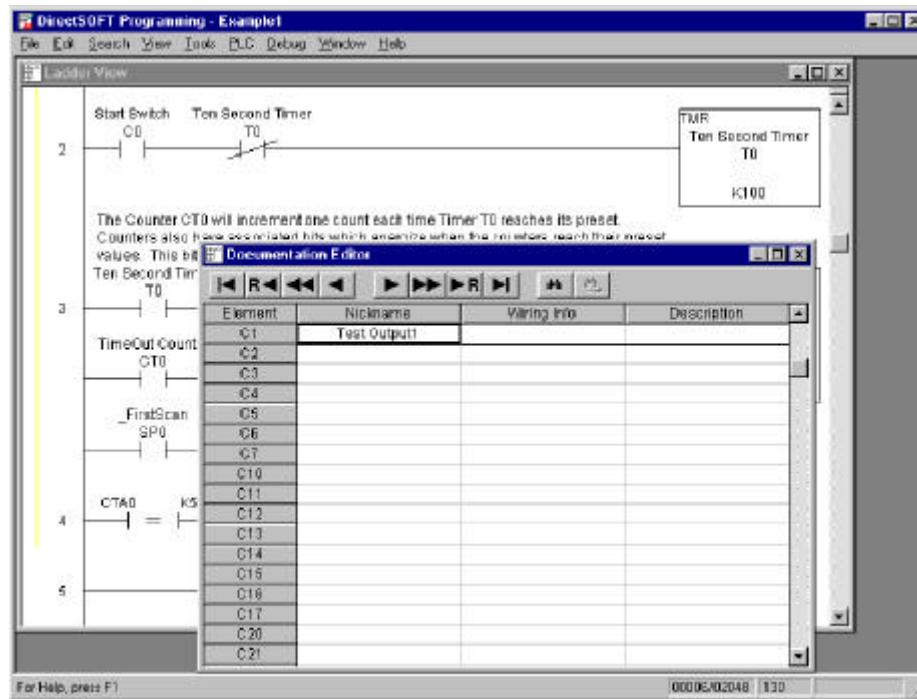
Add **C1** to the rung at this time by moving your cursor to the end of the rung and pressing **F5** to open the **Coil Tab of the Instruction Browser**. Select **Standard Coil** and **OUT** from the available choices. Select **OK** when finished.



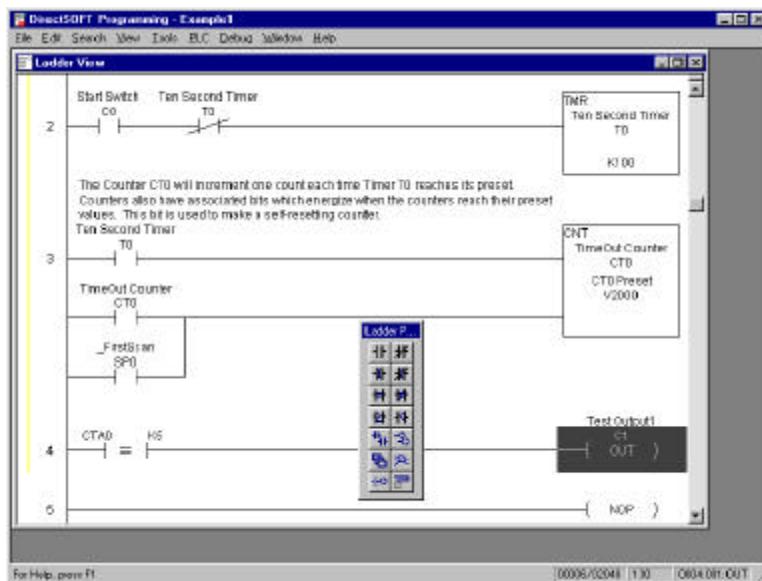
After selecting OK in the Instruction Browser, the Element Dialog box appears. You will be prompted for the output relay designation. In this case, type in **C1**.



You can now enter the nickname “**Test Output1**” for **C1**, using the same procedure used earlier. Press the key combination **CTRL + D** to bring up the **Documentation Editor** and enter “Test Output1” into the **Nickname** window.



Close the Documentation Editor and return to the rung. Notice the Nickname **Test Output1** is now above the element. You are now finished with these rungs. **Accept** them by using the hot key **F8** or by selecting **Edit/Accept** from the upper menu bar.

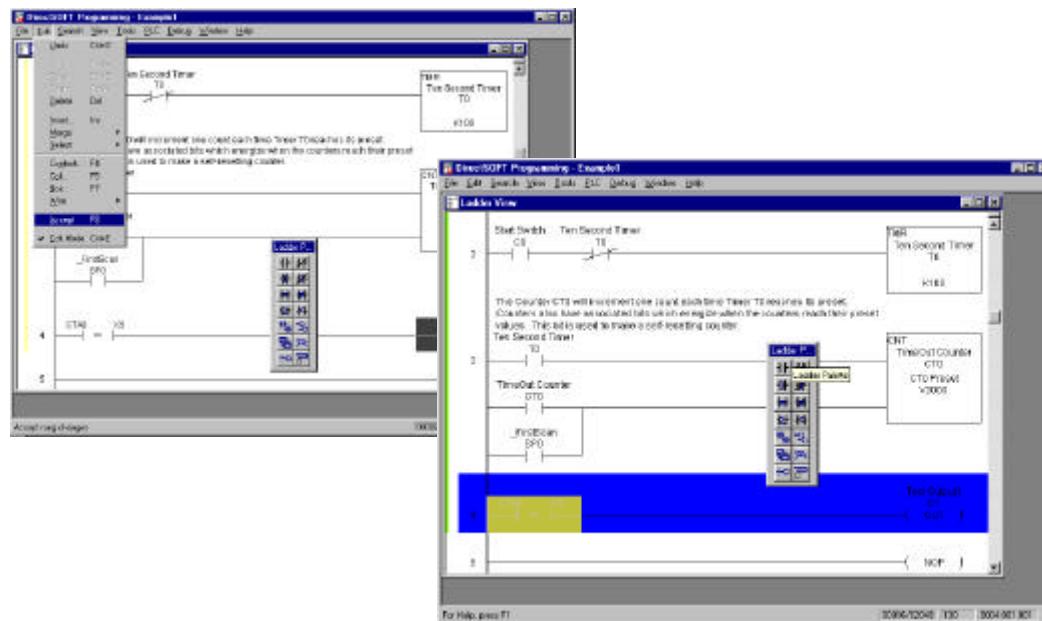


Step 28: Copying to the Clipboard

The following example illustrates the use of the **Copy and Paste** features of **DirectSOFT**. You will copy a rung and paste it to the next. Then, change the count value to **K6** and use output relay **C2** to test it.

To copy a rung, first select the rung. **Rungs cannot be selected and copied unless you have accepted the rung.** Accept the rung by selecting **Edit** and **Accept**. You will see a green vertical bar by the rung when it is accepted.

Now select the rung for copying. This is accomplished by placing your cursor on the rung and using the **SHIFT + Arrow** key combination. With the rung selected, select **Edit** then select **Copy** to send a copy of the rung to the clipboard.



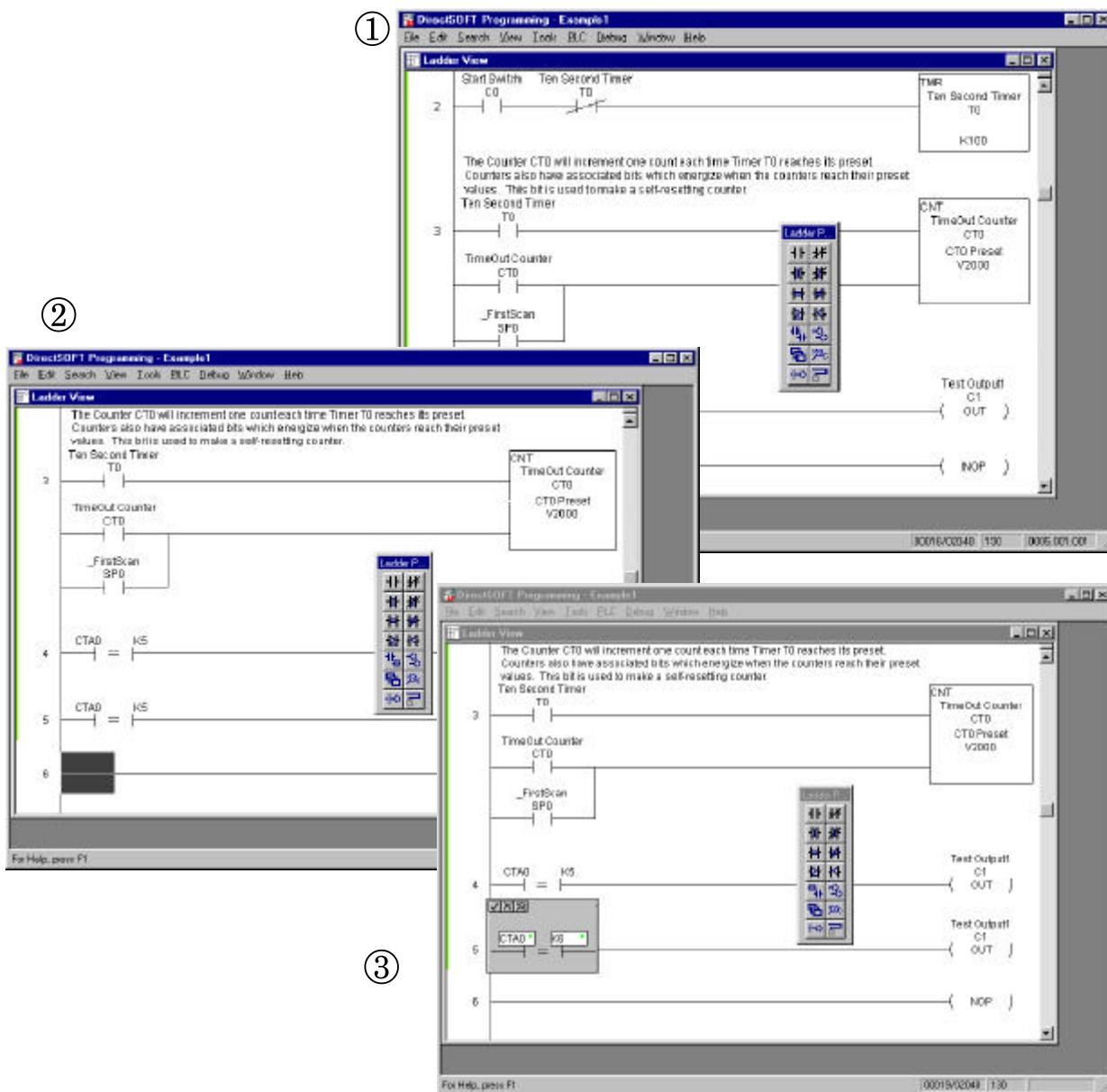
**Step 29:
Pasting to your
Program**

Once you have selected the rung and copied it to the clipboard,
 ① move the cursor down to the next rung in order to paste. The position of the paste
 will always be one rung above the current cursor position.

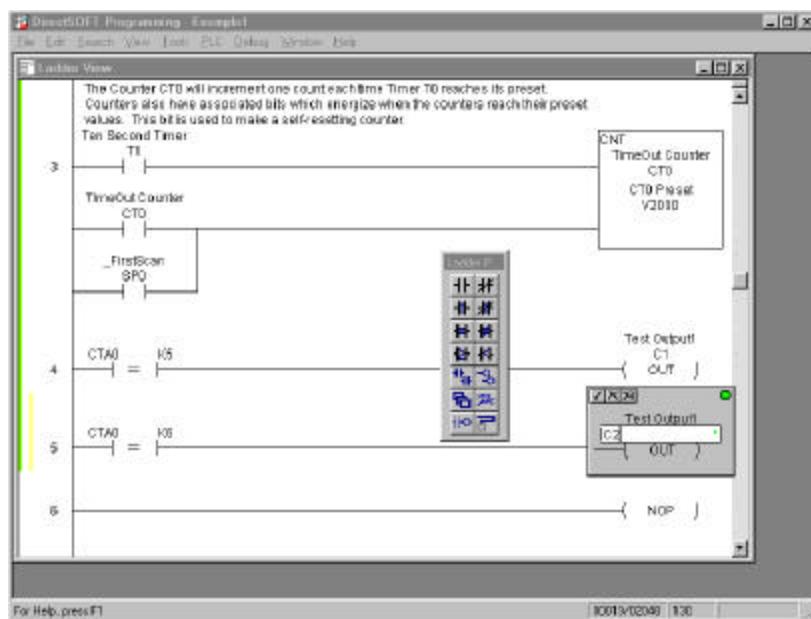
② To paste, select the **Clipboard Icon**, select **Paste** from the **Edit** menu, or
 use the key combination **CTRL + V**. Step 2 below shows the pasted rung in
 position.

Move the cursor up to the pasted rung and start changing the elements. Start by
 editing the conditional contact so it shows **K6**.

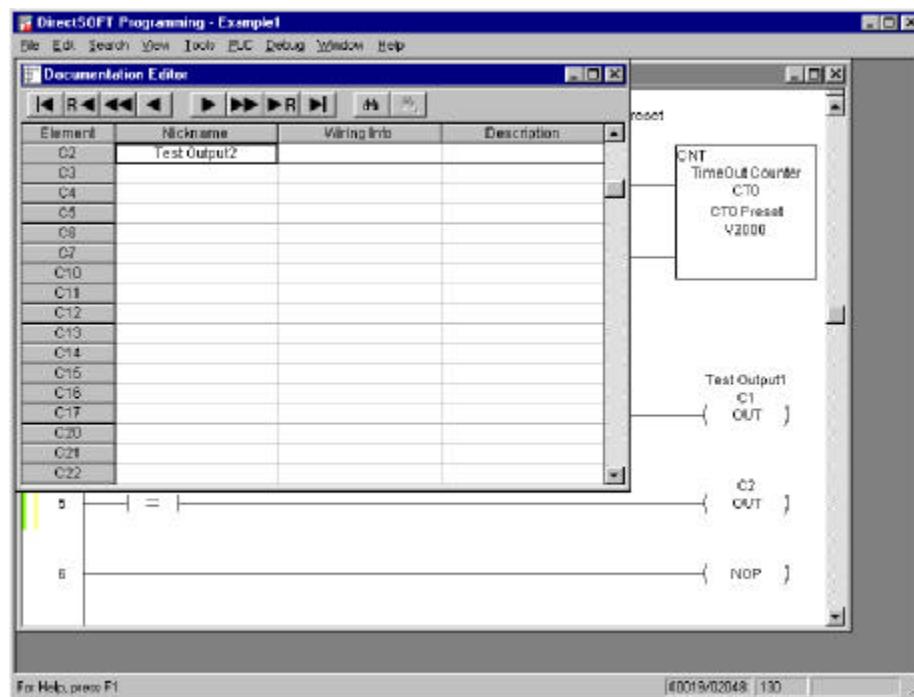
③ When the cursor is on the conditional contact, you can press the **Enter** key
 and the input window will be opened. Press the tab key to move the cursor to the
 right. Type in **K6** in place of **K5** and then select the check mark **✓**.



Next, move the cursor to the end of the pasted rung. With the cursor over the **C1** output element, double click with your mouse. This opens the window for editing the output coil. Change it to **C2**.

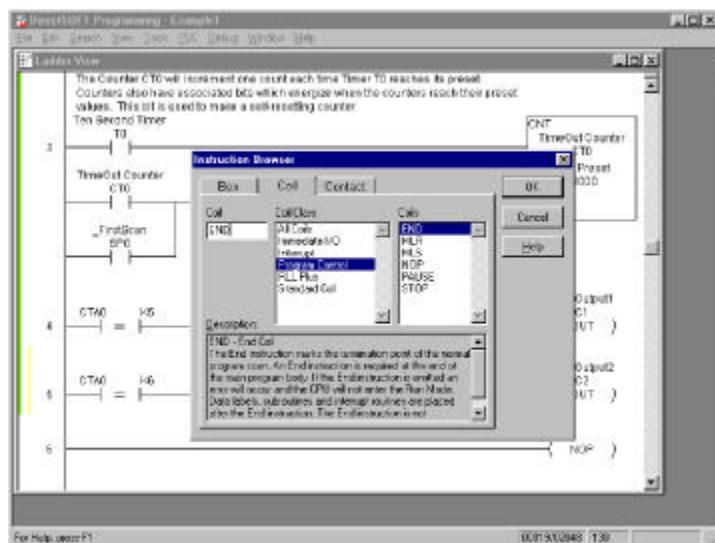


You will also want to assign the nickname **C2** to “**Test Output2**”. Use the key combination **CTRL + D** to bring up the **Documentation Editor**. Follow the steps discussed previously for changing and entering the nickname. When your finished the dialog will show the information given below.

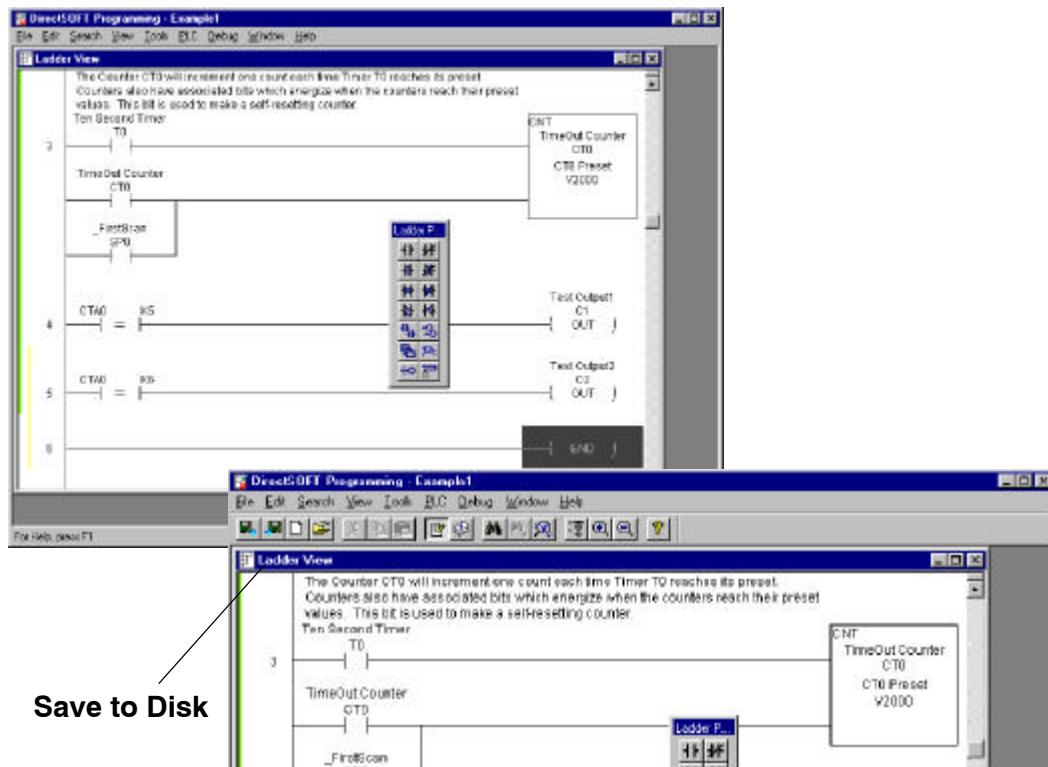


**Step 30:
Ending the
Program**

Every program must have a rung with the **END** command. Move your cursor to the far right of the next rung. Press the **F5** key to bring up the **Coil Tab of the Instruction Browser**. Select **Program Control** under **Coil Class** and **END** under **Coils**.



Click on **OK** when you are finished. With the final rung showing the **END** statement, you have now completed the program. Press **F8** to accept the rungs. Finally, click on the **Save to Disk** icon of the toolbar. You are now ready to connect and communicate with your PLC. Move to the next page and see how to download the program to the PLCs memory and test it.

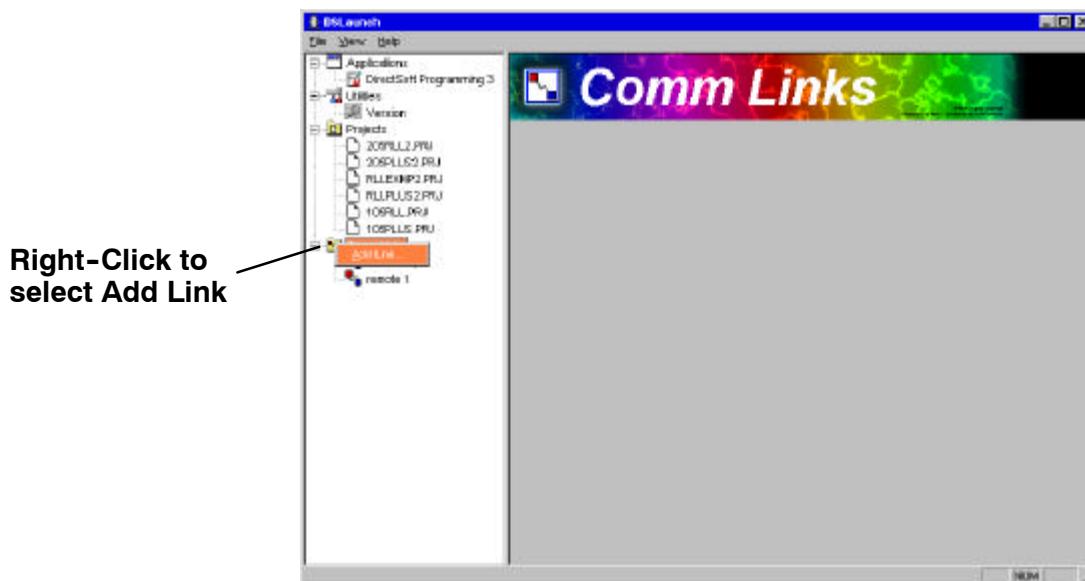


Connecting Online to a PLC & Downloading the Program

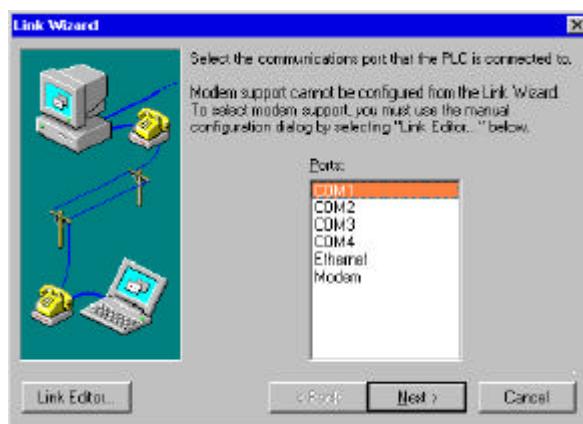
Create a Standard Serial Link

This section will discuss configuration of a Link that uses standard serial ports. If you are creating a serial Link that will connect through a modem, or an ethernet Link, go refer to the Programming Software Users Manual, Chapter 9.

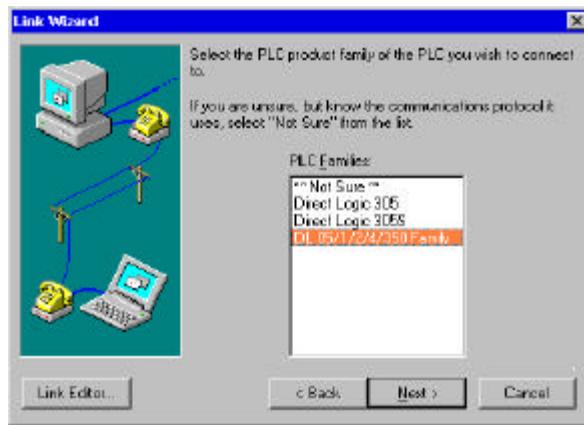
1. The **LinkWizard** can automatically determine the majority of the communication settings. It will search for any existing connections and try to establish the link. If there is no link present or if you want to establish a new link, you can activate the Link Wizard in the Launch Window by **right-clicking** on the **Comm Links** icon and selecting “Add Link”. You can also connect if you are in the **DirectSOFT32** Programming window by selecting **PLC** from the menu and clicking on **Connect....**



2. First, select the port where the PLC is connected. After making your choice, click on **Next >**.



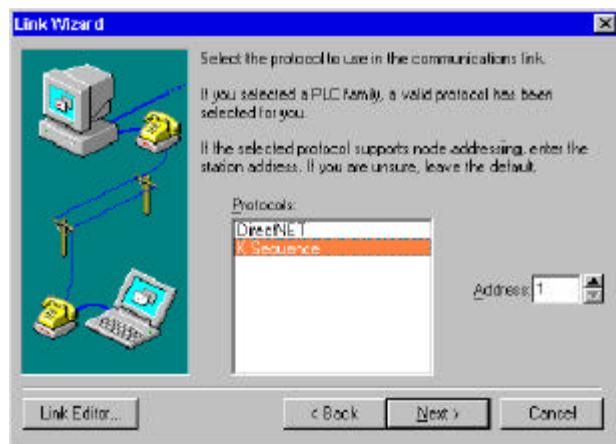
3. Select the PLC family by clicking once with the mouse on the appropriate choice. If you are unsure of the PLC family but know which communications protocol to use, select the “**Not Sure**” choice. If you are using a **DirectLogic** Compatible PLC the LinkWizard can try and detect the model automatically. Click on **Next >** when you are finished.



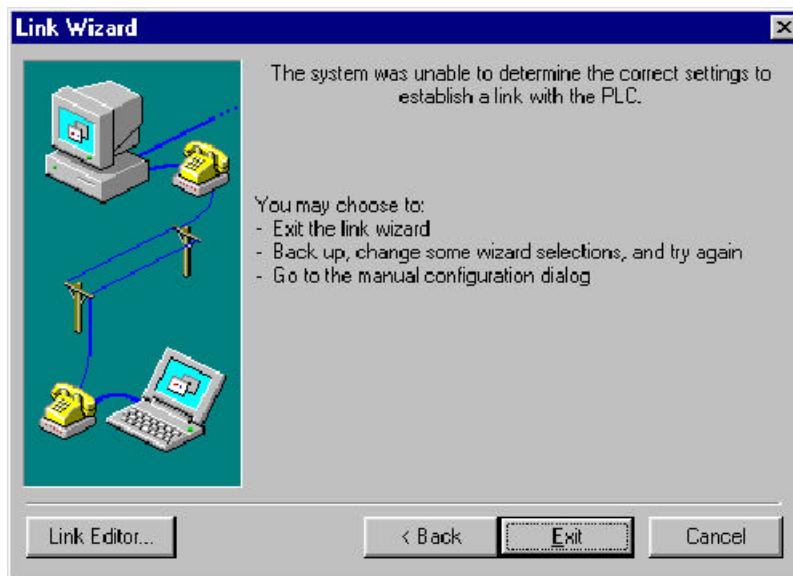
4. Select either **DirectNET** or **K-sequence** protocol. If during the previous step you selected one of the the families listed, the highlight bar will be on a valid protocol for that family. The choice of protocol to use will depend on two factors:

- Whether or not the PLC supports the protocol on the port where you are connecting. See Appendix A for a list of protocols available for ports on **DirectLogic** and compatible CPUs.
- If you need to perform write operations to individual Discrete I/O points or control relays. In this case you must select the K-sequence protocol. **DirectNET** protocol cannot write to individual bit locations.

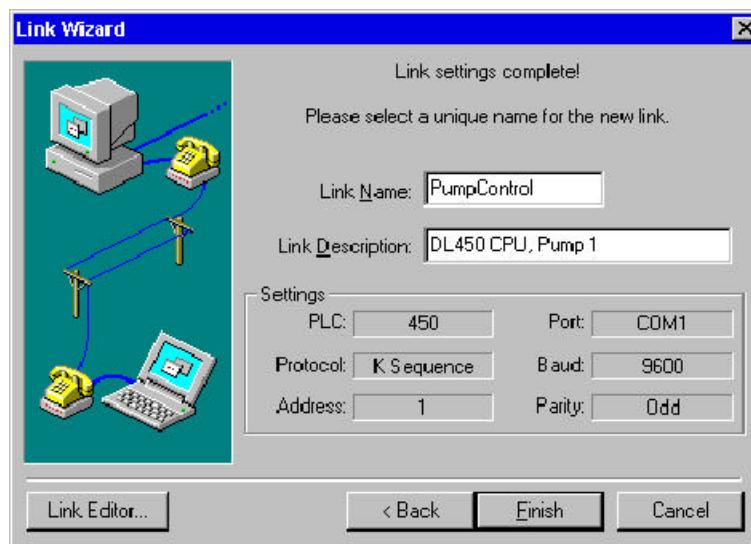
5. If the PLC has been configured to a node Address other than 1, enter that address now. Click **Next >** when finished.



6. **Direct**SOFT32 will attempt to establish a communication Link with the PLC using the node address and protocol you have selected. It will try the combination of 9600 Baud, and Odd Parity. If this combination is unsuccessful, an 'autobauding' sequence will be used to try and determine the correct baud rate and parity combination. If these attempts are unsuccessful, the following dialog is displayed. You can click the **Link Editor** button, and manually attempt to adjust the port configuration, or you can consult the Troubleshooting Guide in Appendix B.



7. If **Direct**SOFT32 is successful in communicating with the PLC, you will be prompted to enter a unique name and description for the Link. Each Link must have a unique name. The name can be up to 16 characters and can contain space characters. The description field allows 32 characters. Enter the name for the link and description then click **Finish** to return to the Setup Links screen.

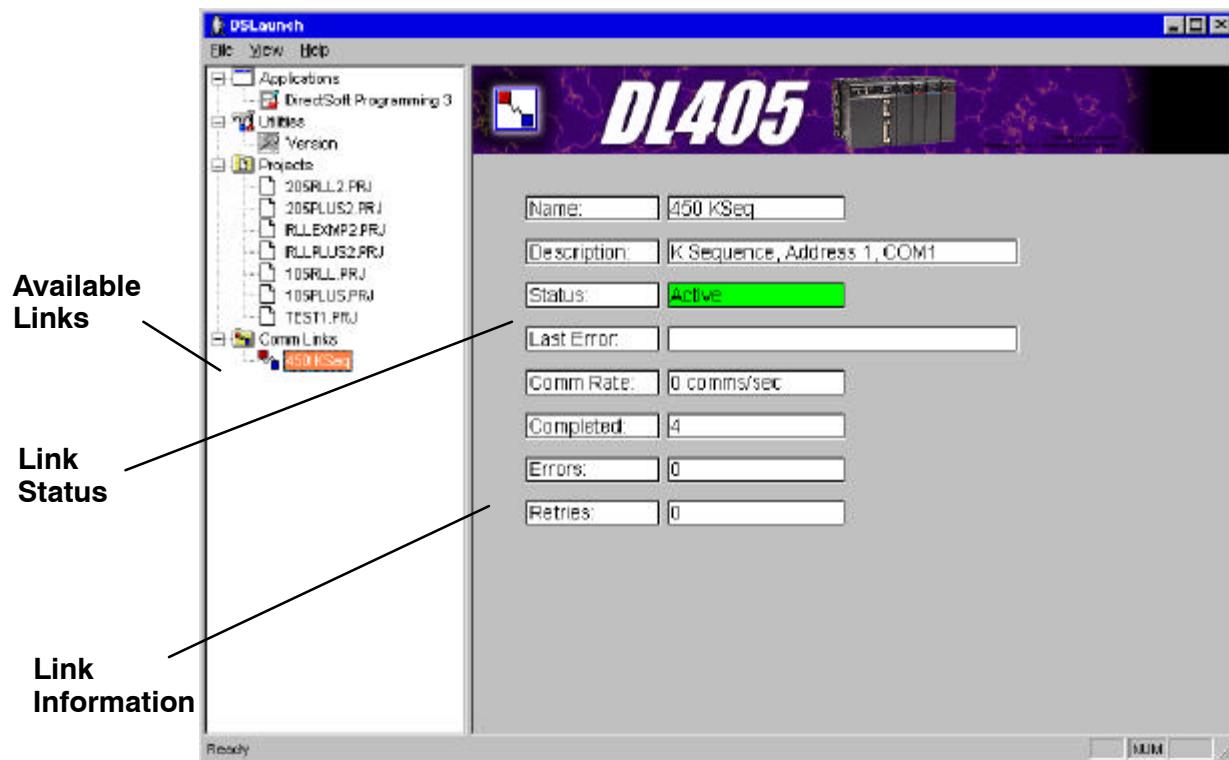


Link Status

After creating a link the name of the link will be displayed in the menu tree under the **Comm Links** icon. When you click on the link all of the configuration information will be displayed on the Launch Window screen. The status field is color-coded to help easily identify the link status.

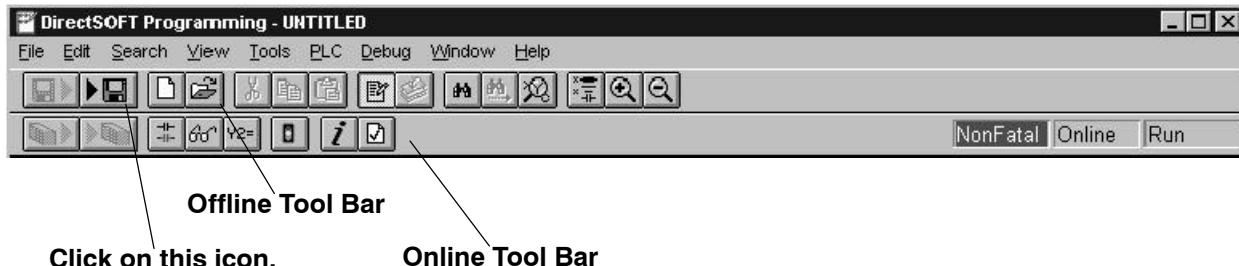
- **Green** — link is already enabled (means it is active and you can use it)
- **Yellow** — paused (You are currently changing the link parameters).
- **Red** — link is disabled (inactive). This does not indicate a problem with the PLC, but that you cannot communicate until the link is active. If a link becomes disabled, **DirectSOFT32** will automatically attempt to enable the link when you “double-click” on the link project.

To go Online with the PLC — “double-click” on the link name with the left mouse button. If there is a program already in the PLC, it will be displayed when the Program Window appears. Below is an example.

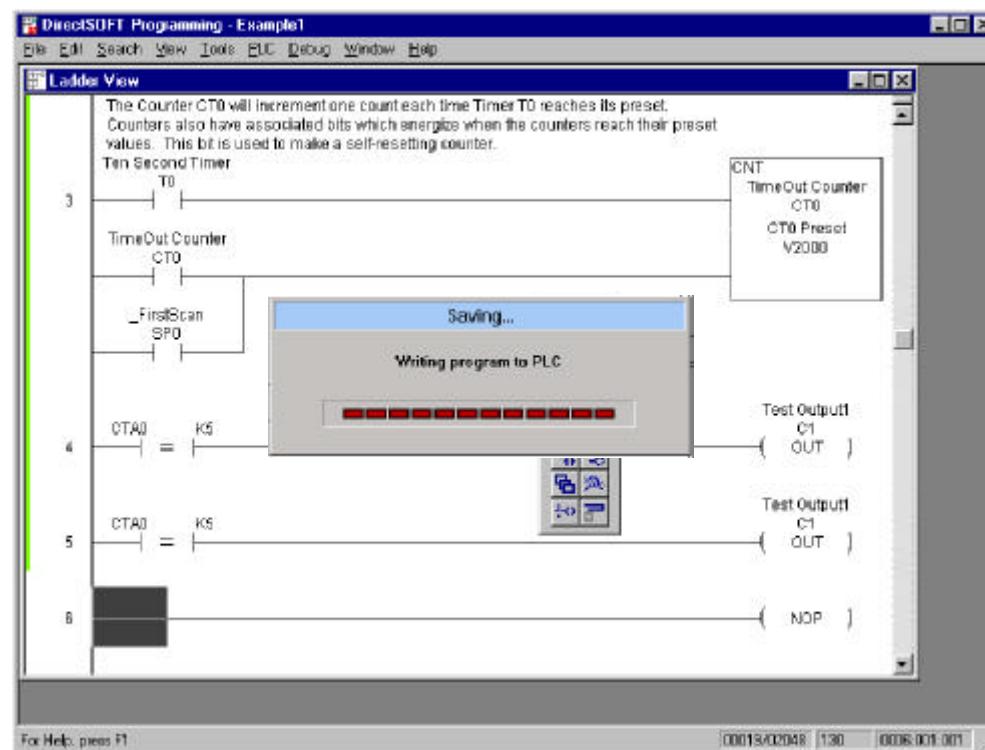


Writing your Program to the PLC

Now that your PC and PLC are properly linked, you can write the program to the PLC. You will note a second tool bar (online tool bar) has appeared below the offline toolbar. The second icon from the left is used to write your program from your hard drive to the PLC. Click on the icon now.



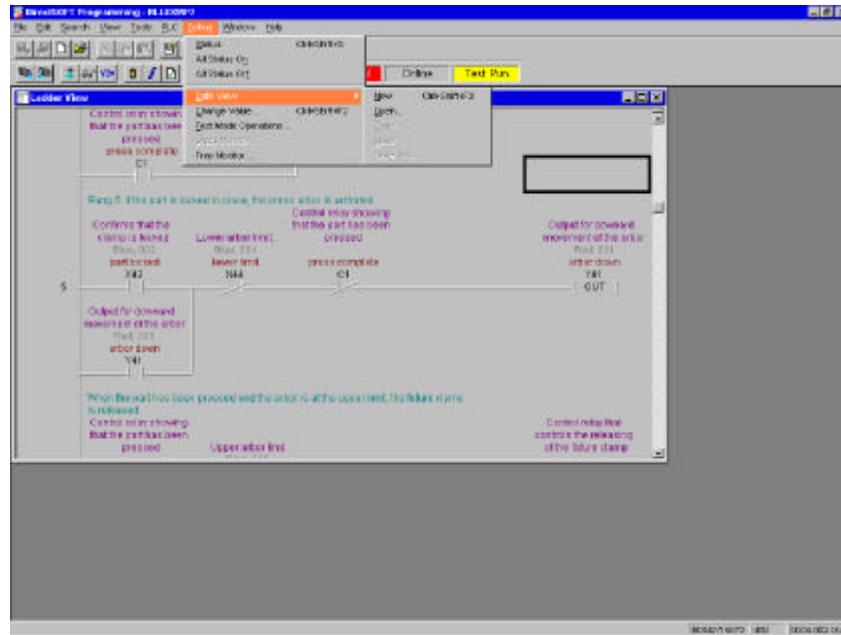
At this point, the program will be saved to your PLC. A window will be temporarily superimposed on your program area. A series of small red indicators will flash in succession to indicate the progress as *DirectSoft32* writes the program to the PLC.



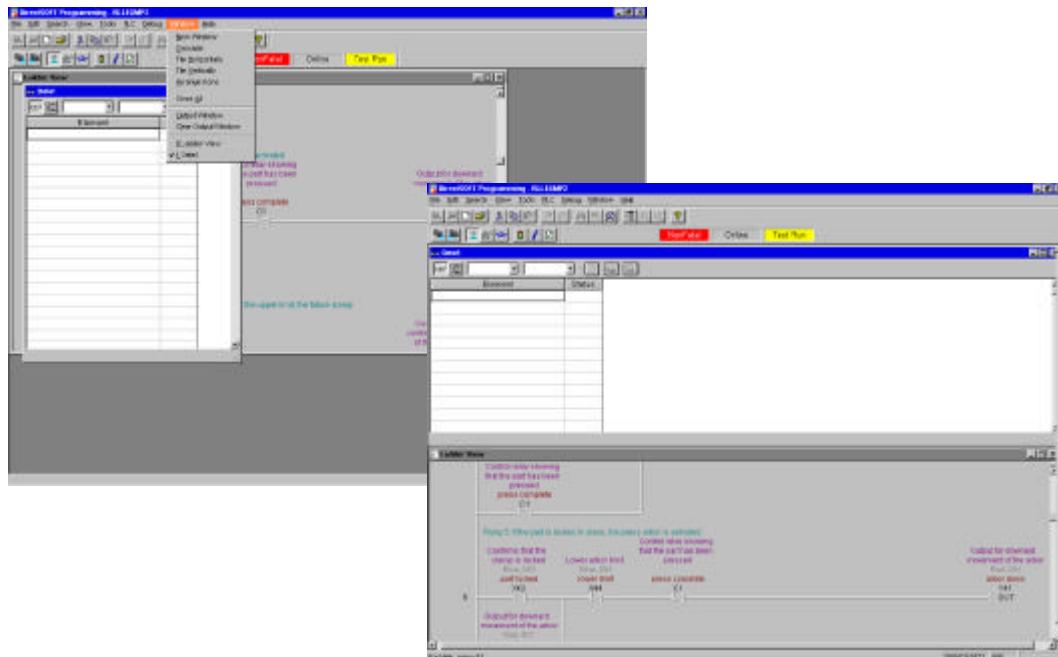
Monitoring the Program with a Data View

Creating a Data View

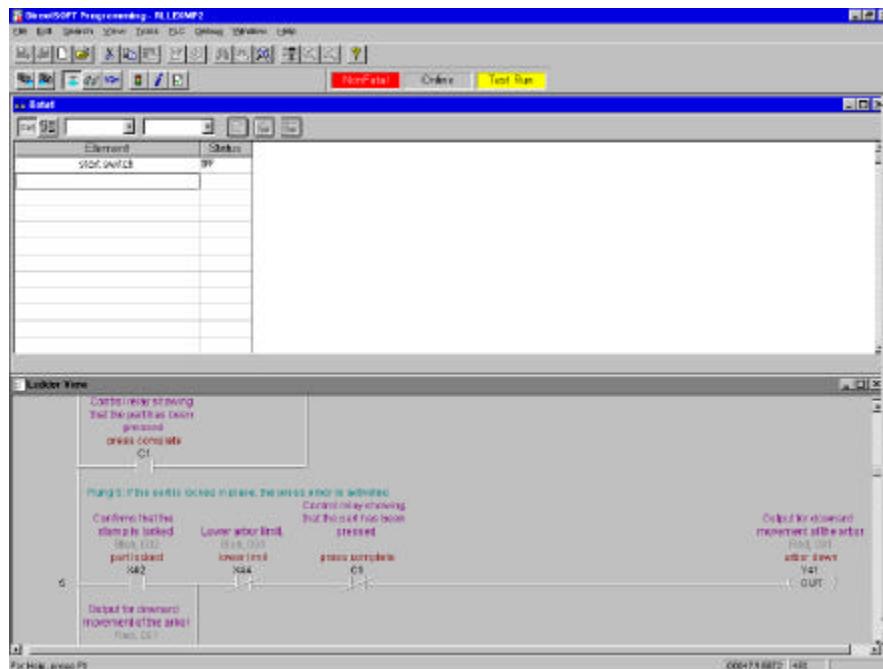
With the program loaded in the PLC, you can now open a **Data View** window to monitor and manipulate the status and data for the various elements of the program. If you have programmed in other languages before, you may know this type of window as a Watch Window. You can access this window by selecting **Debug**, then **Data View**, and **New**.



The Data View window is useful when observed with your ladder logic screen while in the Status ON Mode. To set this up, click on **Window**, then **Tile**.

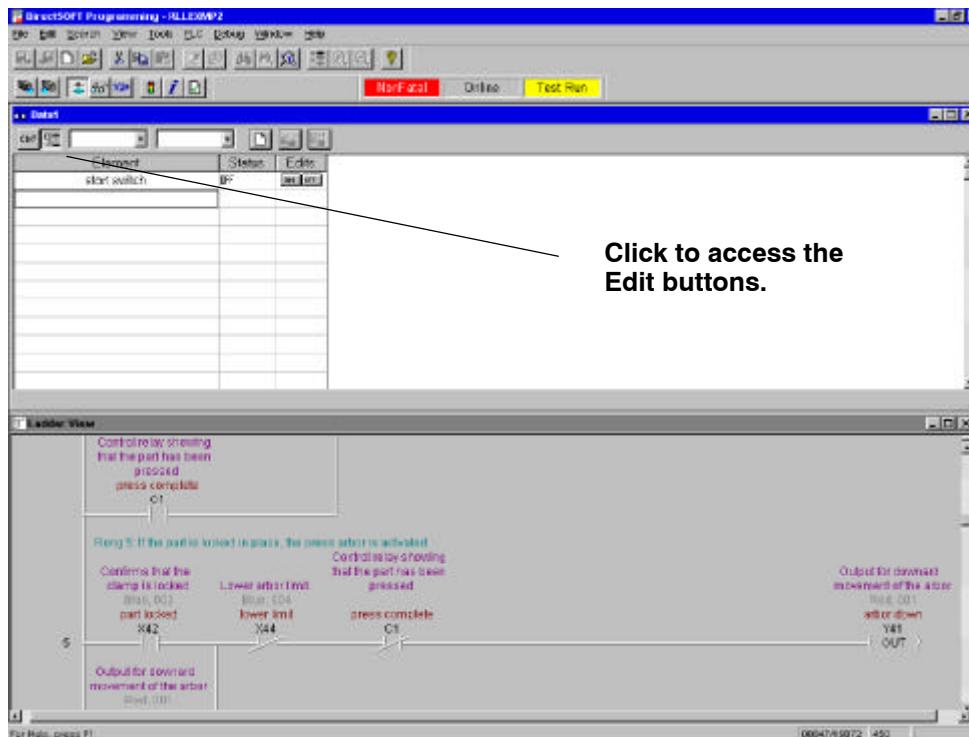


Below the column labeled **Element**, type in **C0** as the first element to be monitored. The software will substitute the assigned nickname, **Start Switch**.

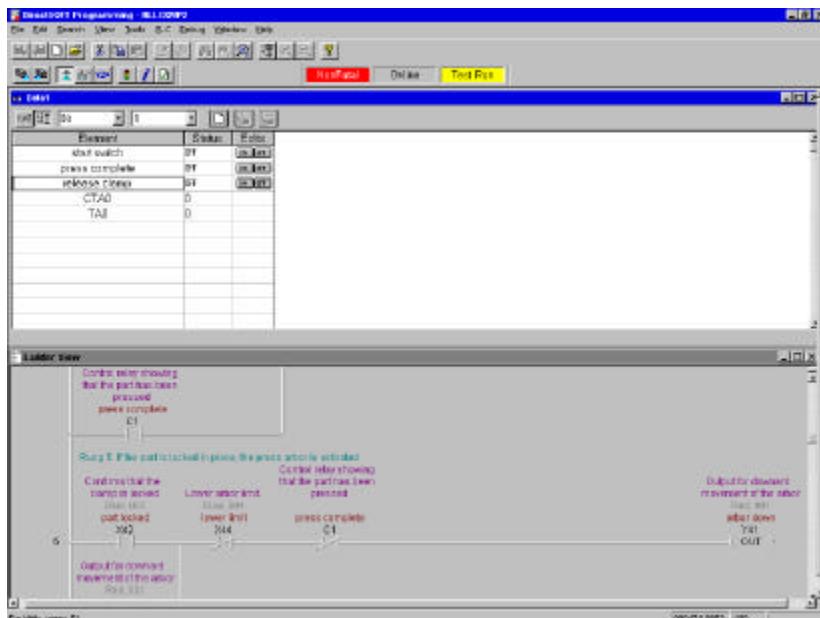


Adding Dynamic ON/OFF Edit Buttons

Now you can setup the Data View window to easily change the status of the observed elements. Notice in the tool bar at the top of the Data View window there is **C1?** and **C1=1 (with an hour glass)**. The **C1?** is active by default, however by selecting the other button, **C1=1**, the ON/OFF **edit buttons** will appear beside the element.

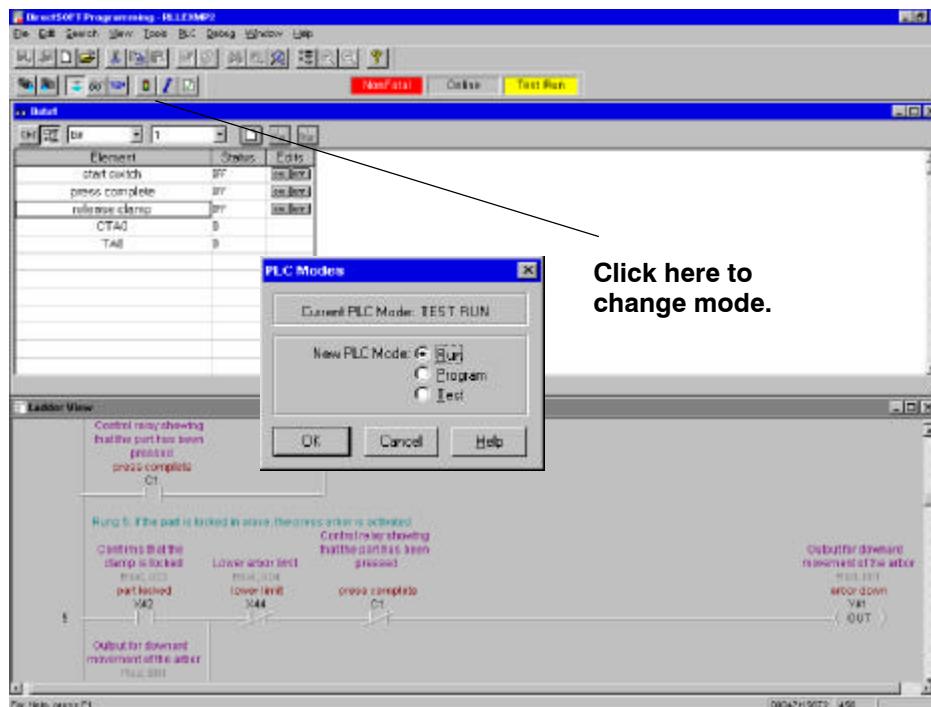


Now you can add the elements **C1 (Output1)**, **C2 (Output2)**, **CTA0 (the counter current value)**, and **TA0 (the timer current value)** to the Data View window. Notice there are no ON/OFF edit buttons for the counter and timer values. This is because you are observing V-memory data for these elements as compared to observing an ON/OFF status.



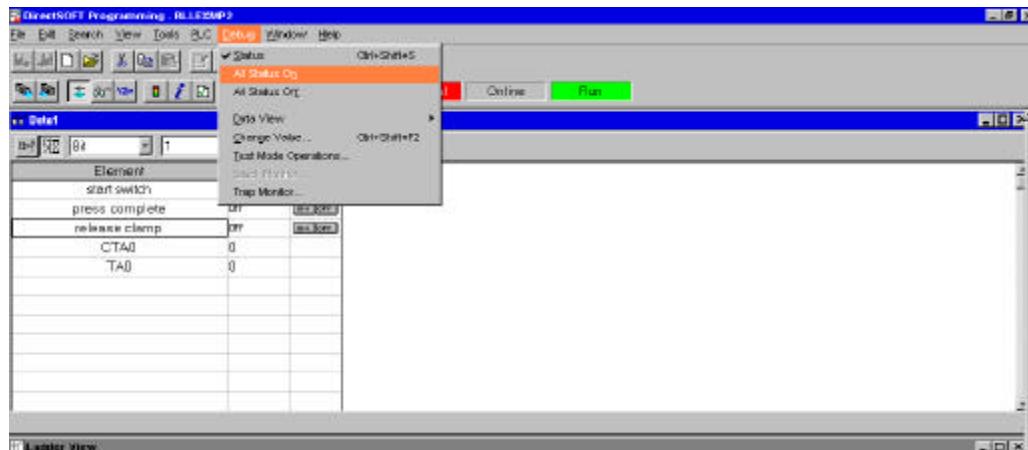
Entering the Run Mode

You are now ready to test the program. First, place the PLC in the **RUN** mode. Select the “traffic light icon” of the on-line tool bar. Then, click on the **Run** mode and **OK**. Alternately, you could have chosen **PLC Modes** from **PLC** of the main menu bar, and then selected **Run** mode from the pop-up dialog box.



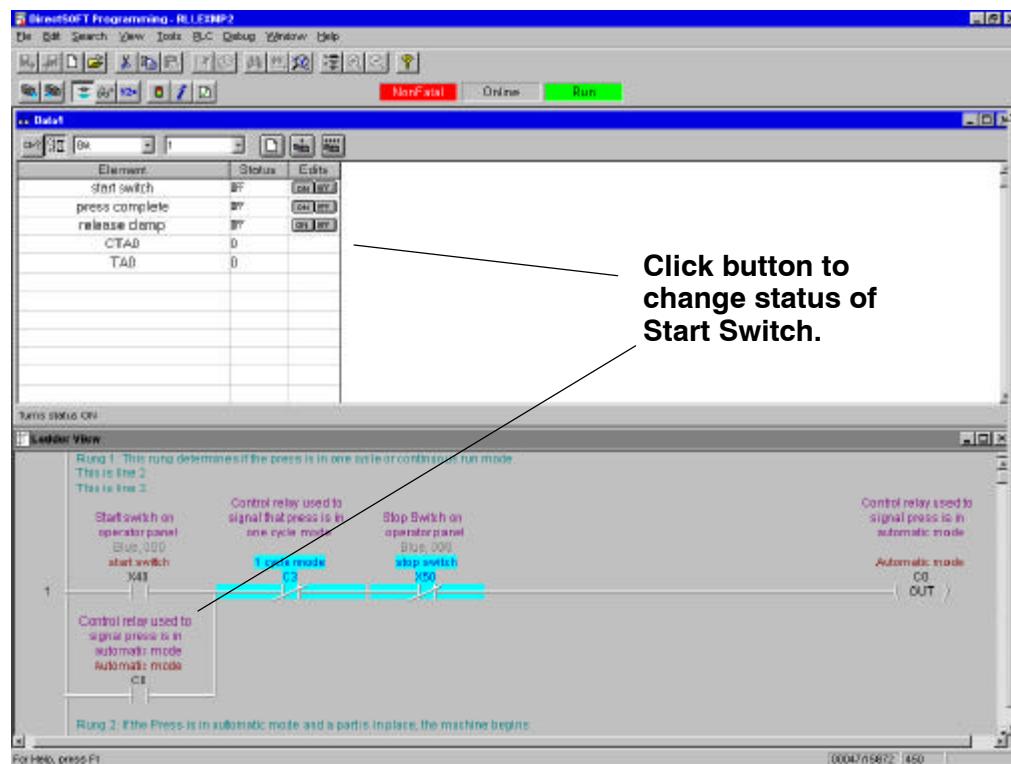
Observing the Status of Elements

To monitor the status of each element, you will need to place the software in the **All Status ON Mode**. Select **Debug**, then **All Status ON**.



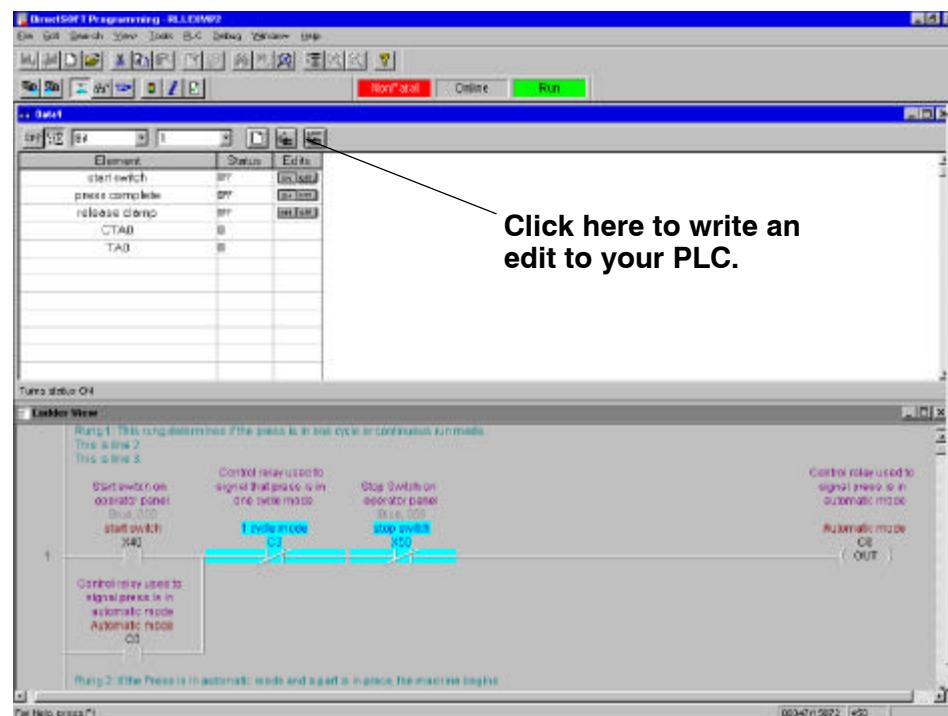
Using the Edit Buttons to Change Status

You can start the program by first clicking the edit button labeled **ON** for the **Start Switch (C0)**. This by itself does nothing. You will need to write a new status to the PLC. The next step shows you how this is accomplished.

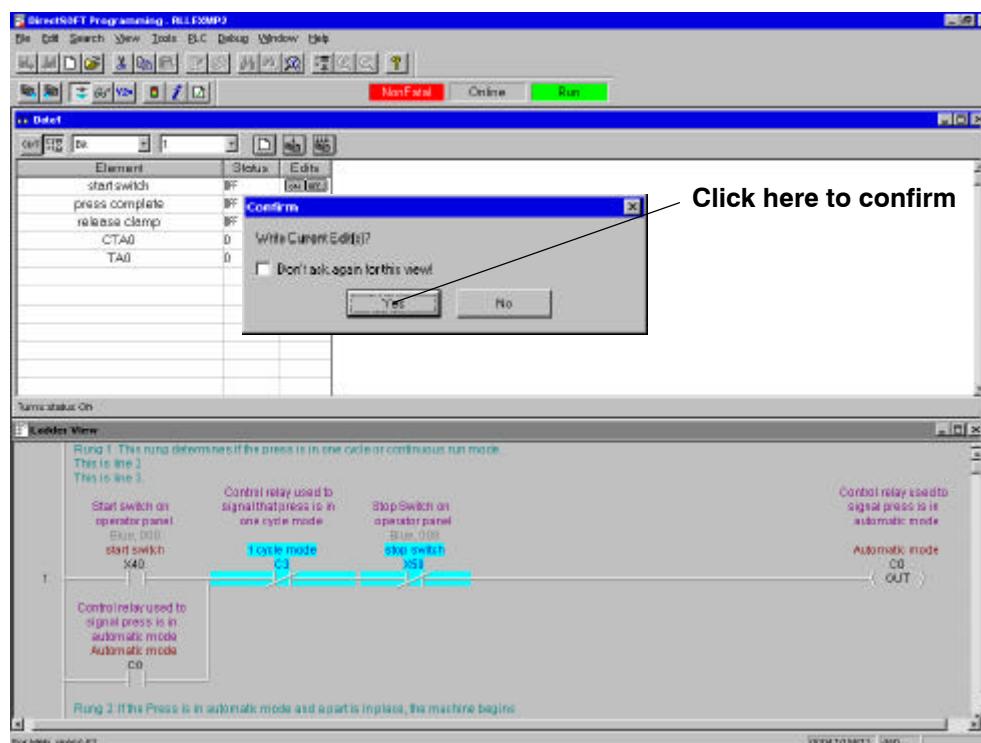


Writing Edits to the PLC

To write the new status to the PLC, select the icon of the Data View (one arrow pointing inward to the PLC).



A confirmation dialog will appear. It will ask if you want to write the edit (or edits) to the PLC. Answer **yes**. The active elements will change color when they are in the ON state. You will see the counter start. Now the program can be observed as it runs.



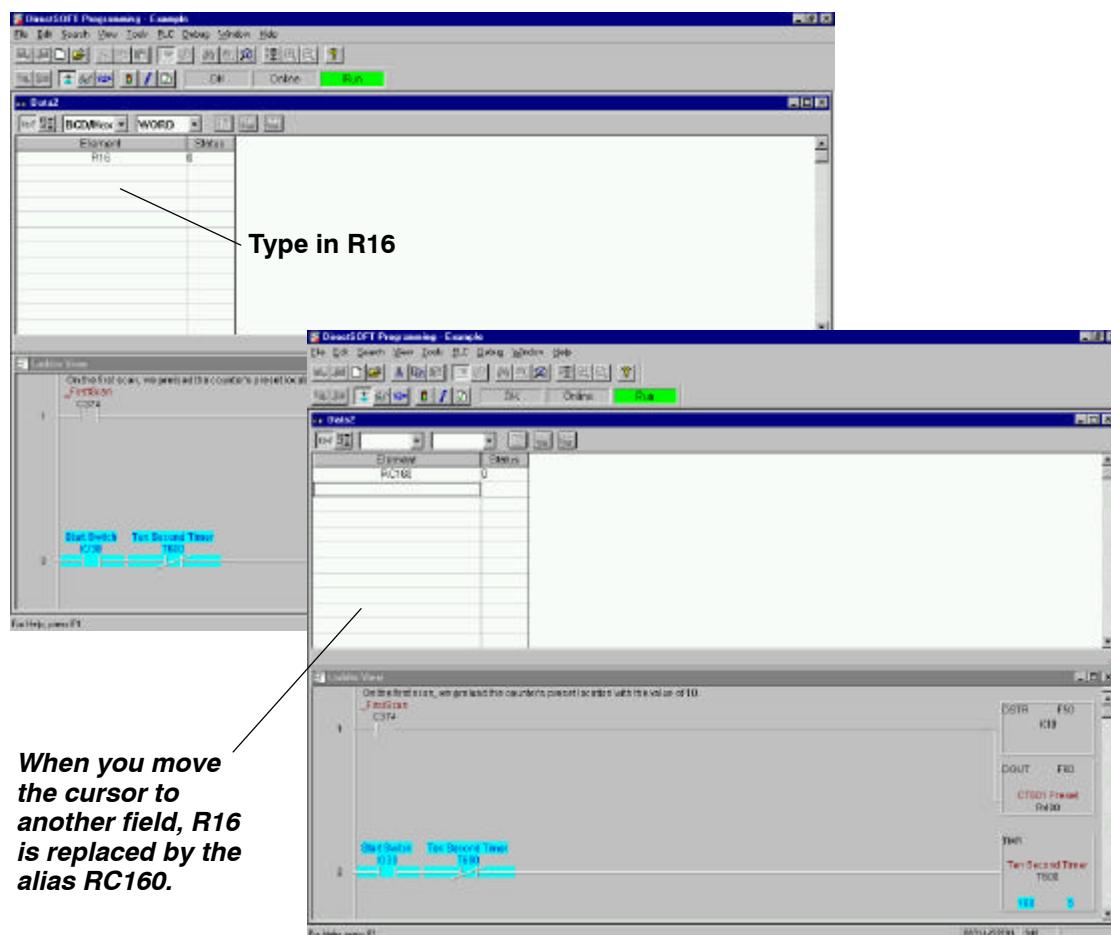
Special Data View Instructions for DL305 Family

DirectNET Restrictions

Unlike the DL105, DL205 and DL405 PLC families, the DL305 family can only use DirectNET protocol. This protocol allows you to READ the status of each internal relay directly, but it will not allow you to use the Data View window to change the status of an internal relay by WRITING to that bit individually. Instead you must do so indirectly by addressing the respective status register that includes the bit whose status you wish to change. Using the example introduced on Page 6 of this manual, C160 was designated as the Start Switch. If you refer to the memory map for the DL305, you would find that C160 is the first bit of status register R16. Consequently, in order to change the status of the Start Switch via the Data View window, you must write an 8-bit word to R16, making sure the first bit is flagged.

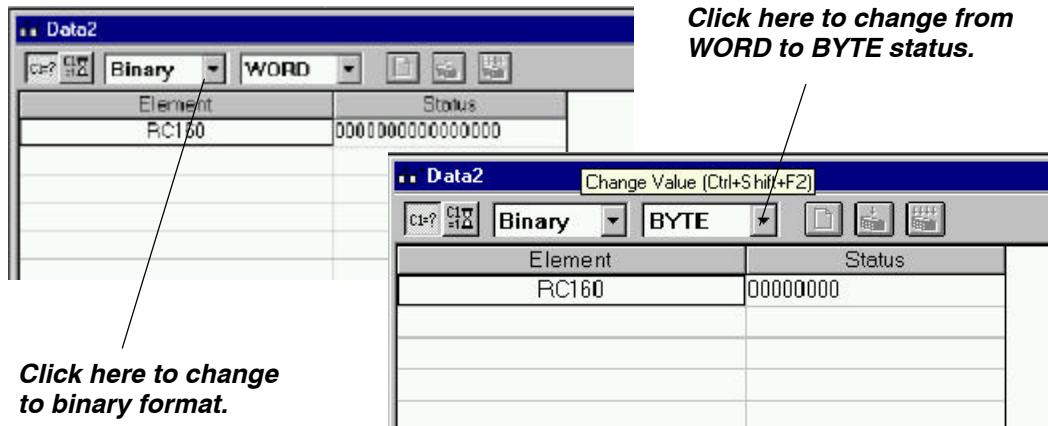
You can start by first entering the element **R16** in the Data View window. When moving the cursor to click another field, you will notice the alias **RC160** appears in place of R16. Do not confuse this alias with C160. The RC designation refers to the entire 8-bit register R16, but C160 refers only to the first bit.

The number shown in the status column, by default, is in BCD/hex format.



Changing the Value to Binary to Observe All 8 Bits. The BCD/hex format does not allow to easily see the status for each of the 8 bits in R16 (RC160). Therefore you will need to change the format to binary.

The Data View window has a “drop down” button that can be used to select the binary format. This button is above the Element column. Select the button and you will see the several choices of formats available. Select **Binary** from the list. Notice now the value shown changes to **0000000000000000**, indicating a binary number format. However, it shows a default of 16 bits. Since the status registers are each 8 bits in size, click on the drop-down button above the Status column and change from **WORD** size to **BYTE** size data. This then displays only eight zeros: **00000000**.

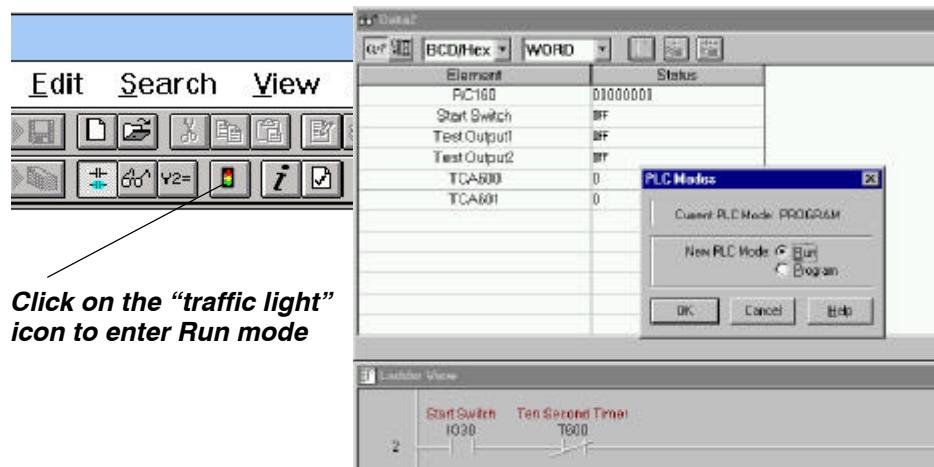


Although you could use R16 to view the status of C160 (Start Switch), C161 (Output1), and C162 (Output2), it is more convenient to view the internal relays directly for read-only purposes. Now you can type in the contacts by name (C160, C161, and C162). The software will substitute the nicknames for you. These use the bit format to show the contacts as being turned ON or OFF. If you see a **1**, the bit is **ON**. If you see a **0**, the bit is **OFF**. Include also **TCA600** (Timer/Counter 600 Current) and **TCA601** (Timer/Counter 601 Current). With the combination of the inclusive status register (R16) and the designated internal relays (C161, C162, and C163) in the Data View window, you can now write to any of these bits via the status register or read them directly by observing their respective bit values (0 or 1). Additionally you can watch TCA600 and TCA601 to see their respective values incrementing as the program runs.

Data2	
Element	Status
RC160	00000000
Start Switch	OFF
Test Output1	OFF
Test Output2	OFF
TCA600	0
TCA601	0

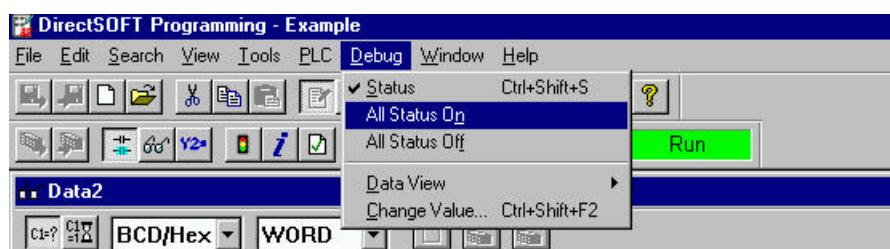
Changing to the Run Mode

You are now ready to test the program. First place the PLC in the **RUN** mode. Select the “traffic light icon” of the on-line tool bar. Then, click on the **Run** mode. You could also have chosen **PLC** from the main menu bar, selecting **PLC Modes** from the list of choices, and then selected **Run** mode from the pop-up dialog box.



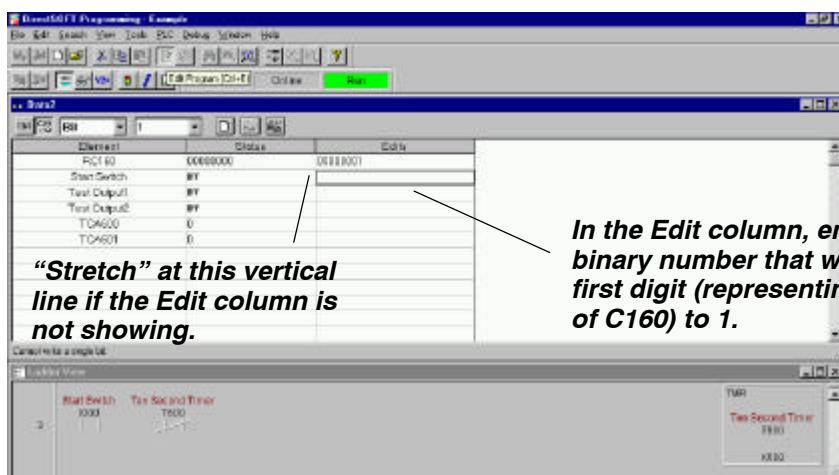
Enabling the All Status ON

To monitor the status of each element, you will need to place the software in the **All Status ON Mode**. Click on **Debug**, then **All Status ON**.



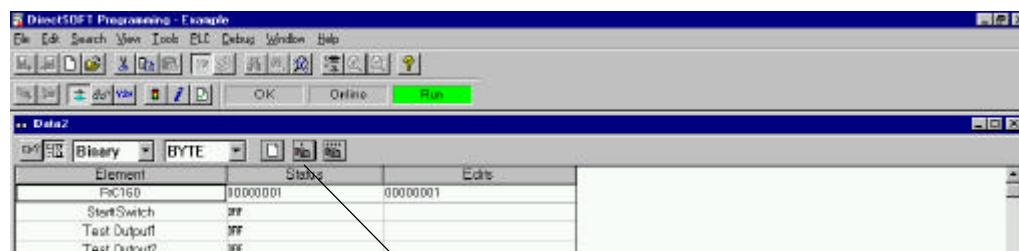
Entering Data for the Status Register

Start the program by writing a **1** to the status of the first bit in R16, which is actually C160. Do this in the **Edit** column of the Data View window. Sometimes the Data View window is not fully expanded. To make the **Edit** column visible, place your cursor on the vertical line that defines the right side of the status label. Hold down the mouse button and drag it to the right. Once the **Edit** column is visible, type in the 8 binary digits with the right-most digit (least significant digit) as a **1**. This is **00000001**.



Writing the Edits to the PLC

To write the new status to the PLC, select the icon with the single arrow pointing down to the PLC from the on-line tool bar . The other icon showing several arrows is for writing multiple edits to the PLC.



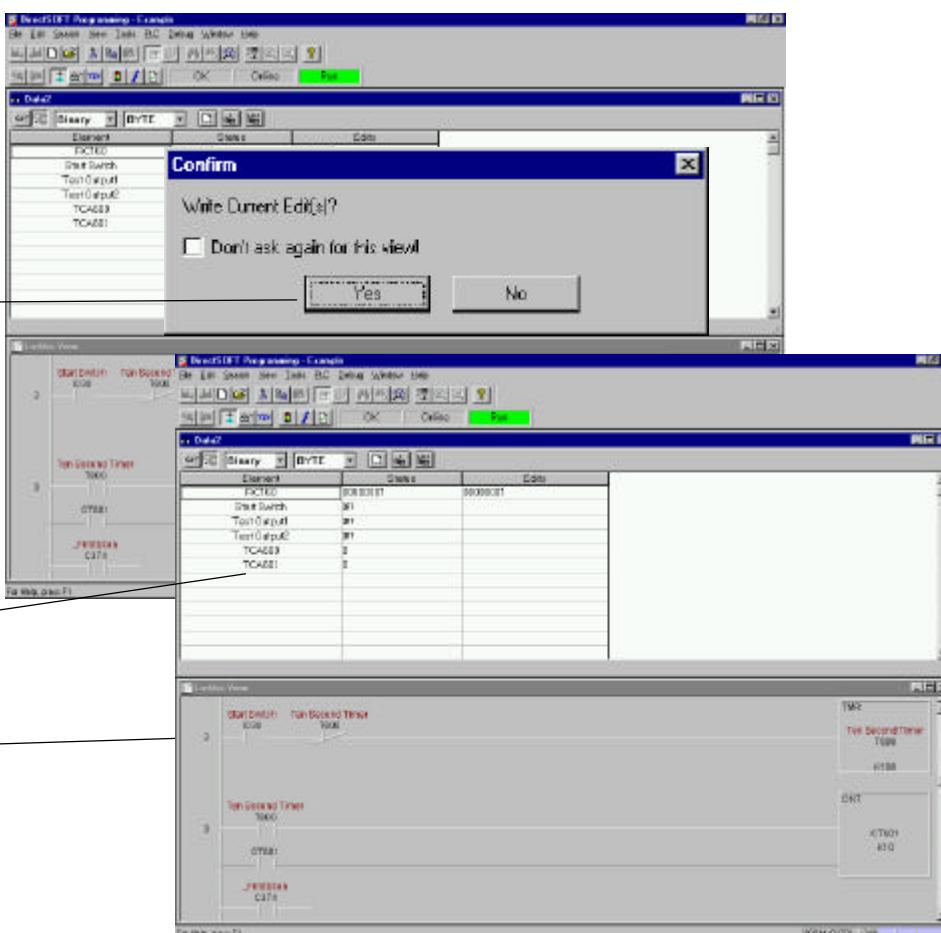
Click here to write your Edit

After selecting the appropriate **Write Edit to PLC** icon, you will then see a confirmation dialog. It will ask if you want to write the edit (or edits) to the PLC. You answer **yes**. With the Start Switch (C160) now ON, you will see the active elements change color as the counter accumulates each tick of the timer and the Boolean conditions are met. Now you can observe how the program runs. Be sure to read the main user manual later.

Click on Yes to confirm.

Timer and counter increments

Start Switch is ON

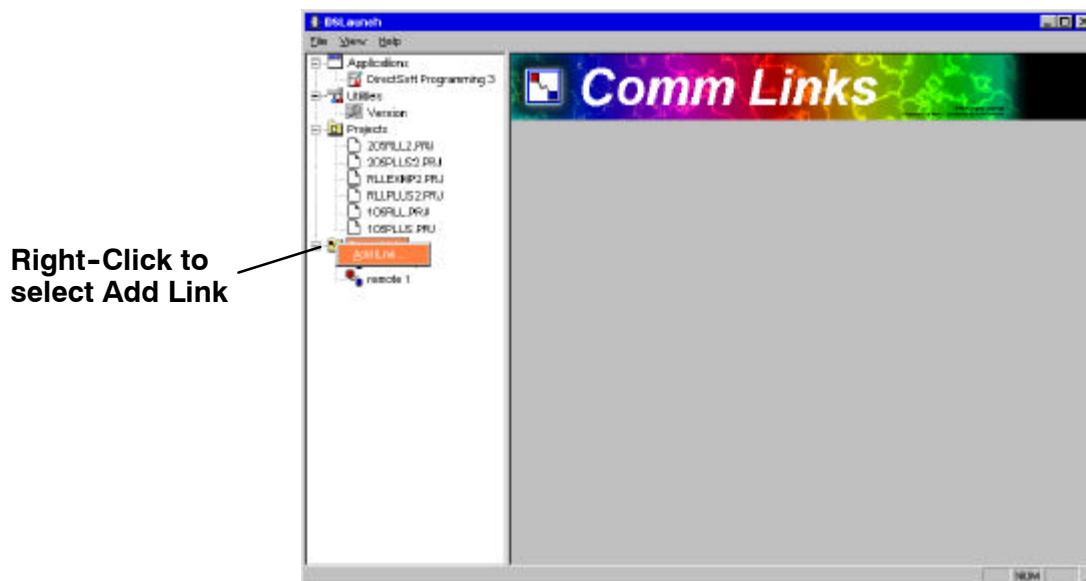


Connecting Online to a PLC & Downloading the Program

Create a Standard Serial Link

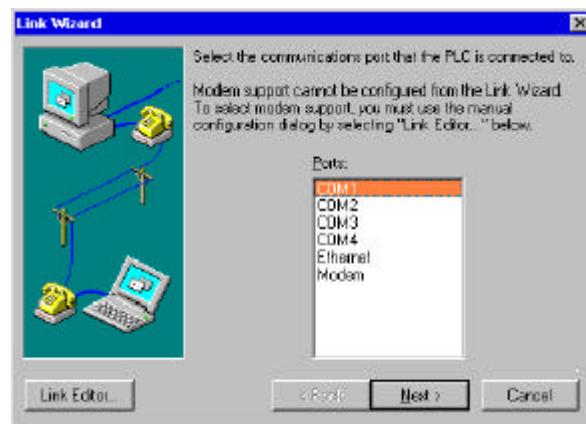
This section will discuss configuration of a Link that uses standard serial ports. If you are creating a serial Link that will connect through a modem, or an ethernet Link, go refer to the Programming Software Users Manual, Chapter 9.

1. The **LinkWizard** can automatically determine the majority of the communication settings. It will search for any existing connections and try to establish the link. If there is no link present or if you want to establish a new link, you can activate the Link Wizard in the Launch Window by **right-clicking** on the **Comm Links** icon and selecting “**Add Link**”. You can also connect if you are in the **Direct**SOFT32 Programming window by selecting **PLC** from the menu and clicking on **Connect....**

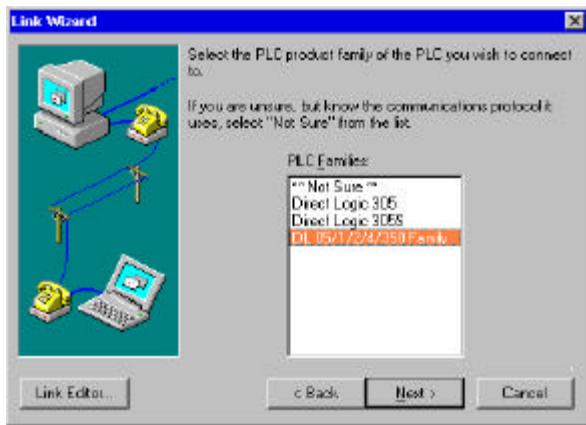


Right-Click to select Add Link

2. First, select the port where the PLC is connected. After making your choice, click on **Next >**.



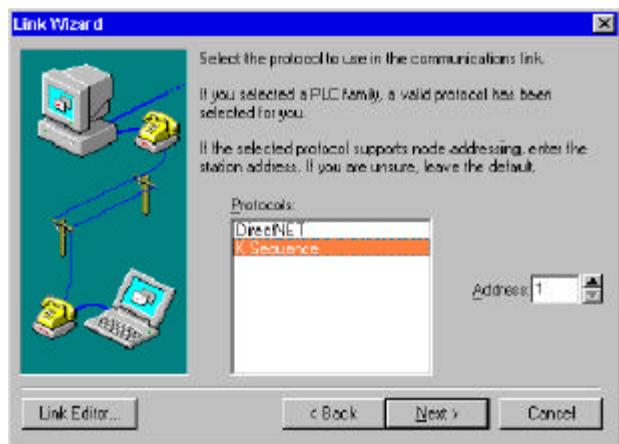
3. Select the PLC family by clicking once with the mouse on the appropriate choice. If you are unsure of the PLC family but know which communications protocol to use, select the “**Not Sure**” choice. If you are using a **DirectLogic** Compatible PLC the LinkWizard can try and detect the model automatically. Click on **Next >** when you are finished.



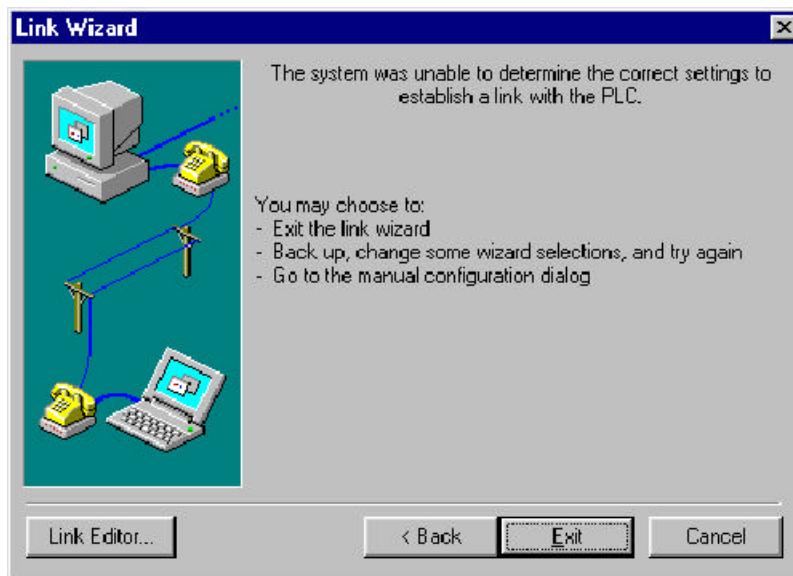
4. Select either **DirectNET** or **K-sequence** protocol. If during the previous step you selected one of the the families listed, the highlight bar will be on a valid protocol for that family. The choice of protocol to use will depend on two factors:

- Whether or not the PLC supports the protocol on the port where you are connecting. See Appendix A for a list of protocols available for ports on **DirectLogic** and compatible CPUs.
- If you need to perform write operations to individual Discrete I/O points or control relays. In this case you must select the K-sequence protocol. **DirectNET** protocol cannot write to individual bit locations.

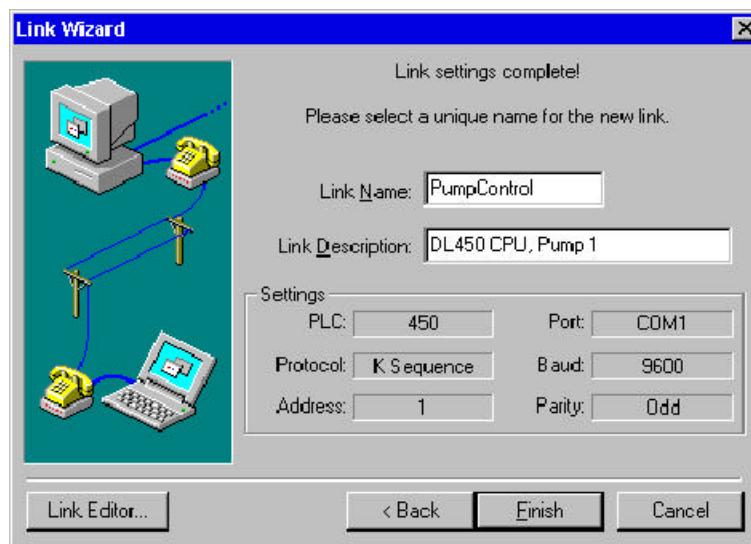
5. If the PLC has been configured to a node Address other than 1, enter that address now. Click **Next >** when finished.



6. **Direct**SOFT32 will attempt to establish a communication Link with the PLC using the node address and protocol you have selected. It will try the combination of 9600 Baud, and Odd Parity. If this combination is unsuccessful, an 'autobauding' sequence will be used to try and determine the correct baud rate and parity combination. If these attempts are unsuccessful, the following dialog is displayed. You can click the **Link Editor** button, and manually attempt to adjust the port configuration, or you can consult the Troubleshooting Guide in Appendix B.



7. If **Direct**SOFT32 is successful in communicating with the PLC, you will be prompted to enter a unique name and description for the Link. Each Link must have a unique name. The name can be up to 16 characters and can contain space characters. The description field allows 32 characters. Enter the name for the link and description then click **Finish** to return to the Setup Links screen.

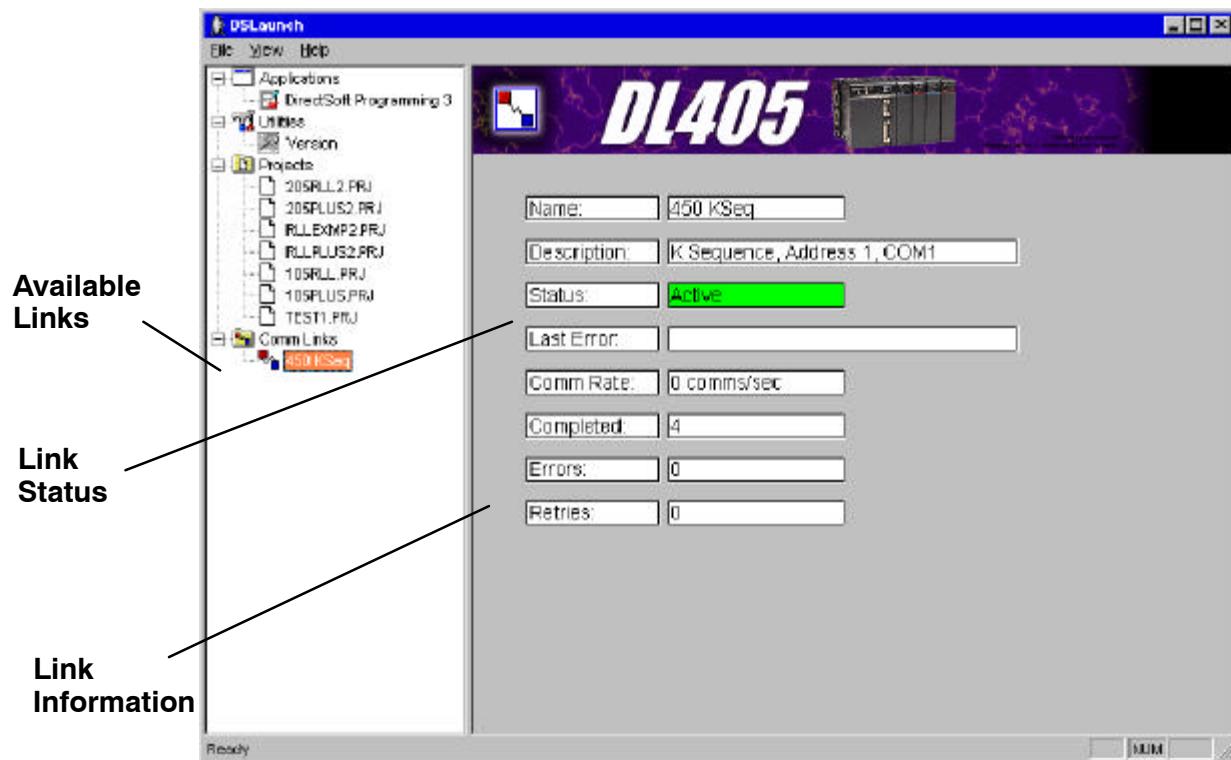


Link Status

After creating a link the name of the link will be displayed in the menu tree under the **Comm Links** icon. When you click on the link all of the configuration information will be displayed on the Launch Window screen. The status field is color-coded to help easily identify the link status.

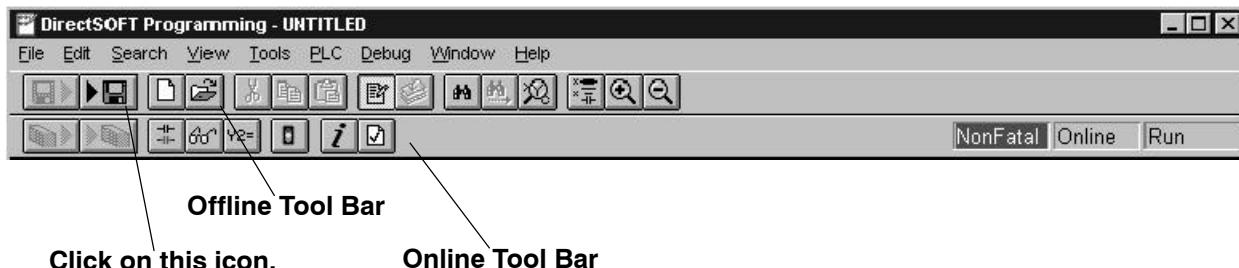
- **Green** — link is already enabled (means it is active and you can use it)
- **Yellow** — paused (You are currently changing the link parameters.)
- **Red** — link is disabled (inactive). This does not indicate a problem with the PLC, but that you cannot communicate until the link is active. If a link becomes disabled, **DirectSOFT32** will automatically attempt to enable the link when you “double-click” on the link project.

To go Online with the PLC — “double-click” on the link name with the left mouse button. If there is a program already in the PLC, it will be displayed when the Program Window appears. Below is an example.



Writing your Program to the PLC

Now that your PC and PLC are properly linked, you can write the program to the PLC. You will note a second tool bar (online tool bar) has appeared below the offline toolbar. The second icon from the left is used to write your program from your hard drive to the PLC. Click on the icon now.



At this point, the program will be saved to your PLC. A window will be temporarily superimposed on your program area. A series of small red indicators will flash in succession to indicate the progress as *DirectSoft32* writes the program to the PLC.

